(2)

University of Pennsylvania
Philadelphia, PA 19104-6390

NEUROMORPHIC OPTICAL SIGNAL PROCESSING AND IMAGE UNDERSTANDING FOR AUTOMATED TARGET RECOGNITION

Technical Report

Prepared for:

Office of Naval Research
Boston Detachment
495 Summer Street
Boston, MA 02110

by

Nabil H. Farhat
Principal Investigator

December 1989

Report No. Eo-MO 89-1

TABLE OF CONTENTS

NEUROMORPHIC OPTICAL SIGNAL PROCESSING AND IMAGE UNDERSTANDING FOR AUTOMATED TARGET RECOGNITION

1. INTRODUCTION

The goal of research described in this report is study of computation and learning in neural net models and demonstration of their utility in image understanding and neuromorphic information processing systems for remote sensing and target identification.

The approach to achieving this goal has two facets. One is combining innovative architectures and methodologies with suitable algorithms to exploit existing and emerging photonic technology in the implementation of large-scale neurocomputers for use in: (a) the study of complex self-organizing and learning systems, (b) fast solution of optimization problems, (c) feature extraction, (formation of object representation), and (d) pattern recognition. The second facet of the approach is to demonstrate and assess the capabilities of neuromorphic processing in solution of selected inverse-scattering and recognition problems. The problem we have chosen to study as test bed for our work is that of automated radar target recognition because of our existing capabilities and expertise in this area.

A summary of accomplishments during this reporting period is as follows:

- Demonstration of the first fully operational optoelectronic or photonic stochastic learning machine (Boltzmann Machine) employing fast annealing by controlled optical injection of noise (noisy thresholding) for optimization and stochastic learning with binary weights.

- Demonstration of neuromorphic target classification and identification from a single look (single broadband echo) employing realistic broadband microwave scattering data from scale-models of actual targets collected in our anechoic chamber radar scattering facility.

- Discovery that most neural net classifiers lack cognitive ability, that is ability to differentiate on their own between familiar and unfamiliar or novel inputs. We have evidence in support of the hypothesis that in order to incorporate cognition, a neural net must be nonlinear and dynamical capable of computing with more than one type of attractor and of bifurcating between different attractors depending on the nature of the input (familiar or novel). When the input is familiar the net computes with one type of attractor and when it is novel it computes with another type and this can serve as mechanism for cognition.

A more detailed description of these findings is given in the next section and in the Appendices.

2. RESEARCH ACCOMPLISHMENTS

Stochastic Learning Machine: In this aspect of our research we have successfully demonstrated what we believe to be the first fully operational optical learning machine (see Appendices I and II for general introduction to photonic neural nets and detail of the Boltzmann Machine). Learning in this machine is stochastic taking place in a self-organizing tri-layered optoelectronic neural net with plastic connectivity weights that are formed in a programmable nonvolatile spatial light modulator (SLM). The net, which can also be called a Boltzmann Learning Machine, learns by adapting its connectivity weights in accordance to environmental inputs. Learning is driven by error signals derived from state-vector correlation matrices accumulated at the end of fast annealing bursts that are induced by controlled optical injection of noise into the network. Operation of the machine is made possible by two important developments in our work: Fast annealing (in approximately 35 time constants of the neurons used) by optically induced noisy thresholding, and stochastic learning with binary

weights which enabled using a binary magneto optic SLM to implement plasticity. Preliminary results obtained with a 24 neuron prototype (8-input, 8-hidden, 8-output neurons) (see pictorial view in Fig. 1) show that the machine can learn, with a learning score of about 70%, to associate three 8-bit vector pairs in 10-60 minutes with relatively slow (60 msec response time) neurons deliberately used to facilitate monitoring evolution of the state vector of the net in time and that shifting to neurons with 1 μsec response time for example, could reduce the learning time by roughly 10^4 times. A subsequent study of methods for improving the learning score show that drastic improvement to a score better than 95% is possible by increasing the number of hidden neurons

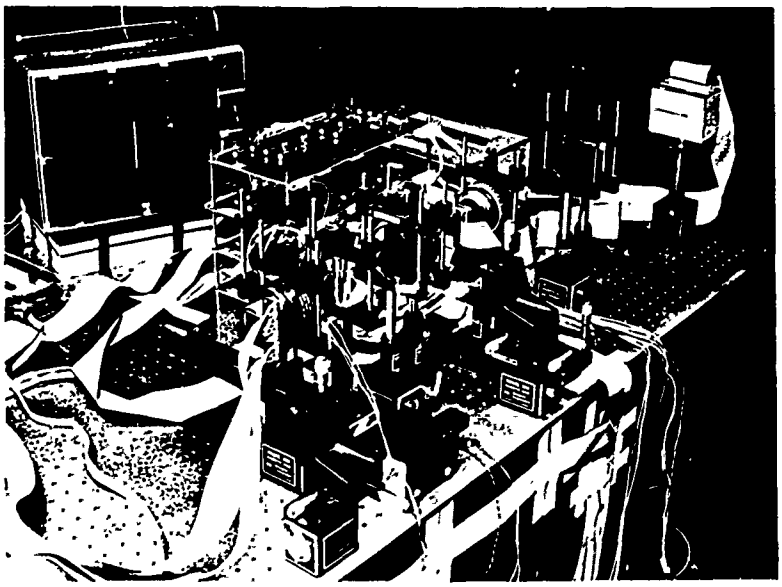


Fig. 1. First fully operational Boltzmann Learning Machine. Methods are under study for compacting this arrangement into clusterable photonic neural chips to enable scaling to larger size nets.

from 8 to 16. Methods for constructing large-scale photonic learning machines of 10^3-10^5 neurons that utilize the concepts developed are under study. It is clear there is an important role for integrated optoelectronics or photonics in the implementation of large-scale neural nets with adaptive learning capability (see Appendix I).

Neuromorphic Radar Target Identification: Past research at the Electro-Optics and Microwave-Optics Laboratory has led to inception and development of *microwave diversity imaging* where angular, spectral, and polarization degrees of freedom are combined to form images of complex shaped objects with near optical resolution. An example of attainable image quality is shown in Fig. 2. This is a projection image of

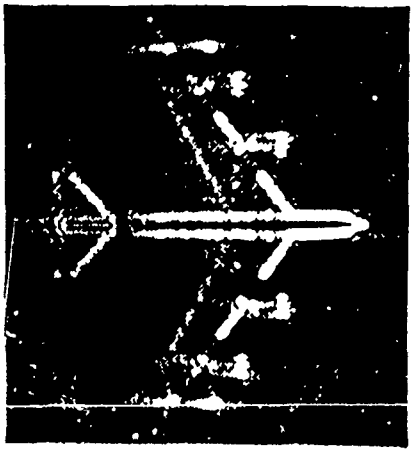


Fig. 2. Microwave diversity image of a complex shaped object

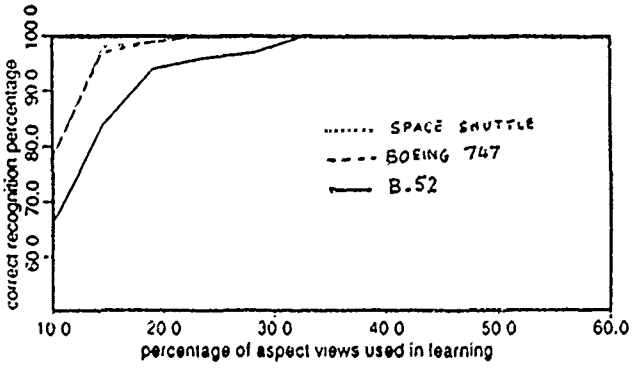


Fig. 3. Learning score (percent or probability of correct identification) vs. size of training set

the scattering centers on a test object (a 100:1 scale model of a B-52). Co-polarized and cross-polarized data sets, each consisting of 128 azimuthal looks (broadband echos) at the target extending from head-on to broad-side (90 degree angular aperture) and an elevation angle of 30 degrees with each look covering a (6-17) GHz spectral window were utilized in obtaining the image shown. Also a novel *target derived reference* technique for correcting the frequency response data for undesirable range-phase (or range-phase time-rate (Doppler) when the target is moving) together with an *image symmetrization* method were painstakingly developed and perfected before the image quality shown in Fig. 2 could be obtained. In later discussion we will be referring to range-profiles of a target. The range-profile at a given target aspect is taken to be the real part of the Fourier transform of the frequency response measured for that aspect corrected for range-phase. For a fixed spectral window and signal-to-noise ratio, the range-profile is independent of range and varies only with aspect.

Application of concepts and methodologies developed and demonstrated in the above research in practice would entail either: (a) use of large, albeit sparse, recording imaging apertures to furnish the angular diversity needed, or (b) use of a single radar system that can track and interrogate a target, in the presence of relative motion, from different aspect angles in time to furnish the required angular diversity in an *inverse synthetic aperture radar (ISAR)* or

spot-light imaging mode. The first approach is prohibitively costly specially when the target is remote and the angular aperture needed to achieve useful resolution is large. The second approach is non-real-time in nature as it requires observing the target over extended time intervals, and this may not be acceptable in numerous applications, in order to synthesize the required angular aperture. One is therefore constrained in practice to limited angular apertures or limited observation times and is thus faced with the longstanding problem of image formation from limited and often sketchy (partial and noisy) information, i.e., one is faced with the classical problem of *super resolution* which has evaded a general solution for a long time. In other words, the problem is to recognize the target from a few looks.

Among its many fascinating capabilities such as robustness and fault tolerance, the brain is also able to recognize objects from partial information. We can recognize a partially obscured or shadowed face of an acquaintance or a mutilated photograph of an acquaintance with relative ease. The brain has a knack for supplementing missing information, based on previously formed and stored associations.

During the period of this report we studied and demonstrated a new concept for automated, distortion invariant (i.e. independent of aspect, range, or location within the field of view), radar target identification from a single "look" (coherent broad-band echo) based on neural net

models and learning. We have explored using a three layered neural net of analog valued neurons with 101 neurons in the first (input) layer, 101 analog neurons in the hidden layer and 2 binary neurons in the third or output (label) layer (to represent three scale models of aerospace targets: space shuttle, Boeing 747 and B-52) and an error driven learning (weights modification) algorithm (error back-propagation (EBP) algorithm). We find (see Appendix III for detail) this net can learn the normalized frequency responses (6-17 GHz window in 101 points) of the target collected for each target in 100 aspect angles ranging in azimuth over 20^{o} extending from head-on towards broadside in such a manner as to be able to classify correctly any one of the frequency responses presented to it by associating it with the correct label. When a two out of three outcomes majority vote is used to designate correct recognition, the learning score is found to be perfect when 35% of the 100 frequency responses of each target are used as the training set (see Fig. 3).

Cognitive Networks: Our research in cognitive networks stemmed directly from the work described in the preceeding section. In that work we find we can make a layered error backpropagation network learn the broadband radar echos (range-profiles) of three test targets (scale models of B-52, Boeing 747, and Space shuttle). The resulting network, which we call adaptive associator network, can generalize very well

and classify the three targets perfectly by triggering one of three associated identifying labels. The classification is robust. It is also distortion invariant, in that the three targets can be identified from a single echos (three echos with majority vote) irrespective of range (scale), orientation, or location within the field of view. (see Fig. 3). The process of learning in this network entails essentially a partitioning of the phase space of the network into three regions each with a fixed point attractor representing one of the three targets. Despite this impressive capability, the network is not cognitive. This means, when presented with echos belonging to a fourth unlearned target, the network responds naively by classifying it as one of the three targets it knows and it is not capable, on its own of indicating that the input is novel. Novelty filters involving front end auxiliary gear which measures other attributes of targets such as size, speed, altitute etc. are frequently proposed as a means for providing additional information that can be used to independently determine whether the target is novel or not before the classification network outcome is considered. Then if the target is novel, the network decision is ignored and if it is not, the classification made by the network is considered meaningful. Obviously the use of novelty filters of this kind is artificial and somewhat contrived. Biological neural networks may use multisensory information

and data fusion to determine novelty or familiarity*, but there is appreciable evidence that they are also endowed with fundamental cognitive abilities which we believe are inherent to the nonlinear dynamical nature of neural structures in the cortex and to the fact these dynamical structures, like all other nonlinear dynamical systems, compute with three types of attractors: fixed (limit point), periodic (limit cycle), and chaotic and that bifurcation between these types of attractors may play a role in cognition, as our preliminary findings suggest. Bifurcation may also be important in hierarchial processing and in higher order functions produced by these structures. It is intriguing to consider that a chaotic attractor is an information machine, in the sense that one can not predict the next state of a chaotic network given its present state, and that bifurcation between chaotic and periodic attractors has been observed in the olfactory cortex of the rabbit and proposed as a possible mechanism for odor identification [1].

Networks that bifurcate under the influence of environmental input between chaotic and periodic attractors may be endowed with richer behavior than those bifurcating between periodic and fixed point attractors. Their numerical simulation and study is however more involved than that for periodic attractor networks. Our study of

*Most probably multisensory information is being used for some sort of supervised learning.

cognitive networks and their application is therefore focusing initially on the easier case of periodic/fixed point attractor networks. The insight and experience gained with these networks will then be applied to the study of chaotic networks. The ultimate aim of this research is to devise methods for controlling the phase-space behavior of neurodynamical systems (phase-space Engineering) and to demonstrate the power of cognitive networks in pattern recognition in general and in ATR in particular (see Appendix IV). The ATR problem is what we consider a "convincing application" for neural networks. We know that the problem of target identification from a single broadband echo has so far resisted solution by conventional means. Demonstrating that a learning cognitive network can be successfully used for robust automated target recognition will be an important achievement. It is a challenging task which will help establish the viability of neurocomputing in an objective manner. Another reason for selecting the ATR problem as test bed for research in cognitive networks is the extensive experience and measurement facilities we have accumulated in this area which allows us to work with realistic electromagnetic scattering data representing scale models of actual targets of interest.

A future goal in this aspect of our research is therefore to incorporate cognition in neurodynamical system through synchronicity in cognitive dynamical bifurc ng networks that compute with diverse attractors comb ch

clustering networks that can handle multisensory information and compute with fixed point attractors. More specifically we are starting to investigate the feasibility of clustering the echos from a given target into M labels which are stored in one isolated periodic attractor (a closed version of the string or sequential attractor described in Appendix IV) of the cognition network. This would be done for each target the composite network is required to recognize. The periodic attractors of the individual targets stored in the cognition network will be highly isolated and not intersecting and each will have an imbedded label identifying its target. The use of multisensory information such as range-profile data (derived from frequency response data) and polarization information for example will be studied as means for resolving if required any errors in the performance of the clustering network arising from ambiguities between the range-profiles of different targets.

3. CONCLUSIONS

To realize the potential advantages of neuromorphic processing, one must contend with the issue of how to carry out collective neural computation algorithms in real-time at high speed exceeding speeds possible with electronic digital computers. Obviously parallelism and concurrency are essential ingredients and one must contend with basic implementation issues of how to achieve such massive

connectivity and parallelism and how to achieve artificial plasticity, i.e. adaptive modification of the strength of interconnections (synaptic weights) between neurons which is required for implementing memory and learning (self-programming).

The answers to these questions seem to be coming from two directions of research. One is connection machines in which a large number of digital central processing units are interconnected to produce parallel computations in VLSI hardware, the other is analog hardware where a large number of simple processing units (neurons) are connected through modifiable weights such that their phase-space dynamics has associated with it useful signal processing functions. The concurrent digital processing approach provides flexibility but has to contend with the communication overhead between individual processors which appears to limit presently the number of modifiable connections per second in simulated networks to 10^6 or 10^7. No such communication overhead is associated with the fine grain neural approach where the processing elements carry out a simple operation.

Analog photonic hardware implementations of neural nets [2],[3], since first introduced in 1985, have attracted considerable attention of the optical processing community for several reasons. Primary among these is that the photonic approach combines the best of two worlds: the massive interconnectivity and parallelism of optics and the versatility, high gain, and decision making capability

(nonlinearity) offered by electronics. Ultimately it would seem more attractive to form very large analog neural hardware by completely optical means where switching of signals from optical to electronic carriers and visa versa is avoided. However, in the absence of fully optical decision making devices with versatility comparable to that offered by optoelectronic amplifiers, the capabilities of the photonic approach remain quite attractive and could in fact remain competitive with other approaches when one considers the flexibility of architectures possible with it and its potential for realizing more biomorphic and complex neurons of the type needed for neurodynamical spatio-temporal networks as will be explained below.

The photonic approach is based on dividing machine functions into two parts. One is a programmable optically interrogatable synaptic plane (connectivity mask) for storing the values of connectivity weights between the active elements (neurons) of the system. The weights would be down-loaded from a computer controller via either electronic interface or optical interface to the mask. The connectivity mask would be completely reconfigurable and would furnish therefore not only programmable weights, but also alterable topology or architecture. Thus any number of layers with feedforward and/or feedback would be possible. The ability to dynamically change the topology of the network is important for the study of new learning algorithms that call for adaptive topology as potential means for overcoming N-P

completeness of learning. The second part of the machine consists of an array of programmable analog amplifiers that furnish the required nonlinear neuron response. This is basically the approach adopted in the Boltzmann learning machine described in this report.

The parallel optical readout of the synaptic plane is the one distinction of this photonic approach as compared to an entirely electronic LSI or VLSI implementation where all operations must be carried out serially. Although the weights both in the photonic approach may be loaded serially, computing both the activation potentials $u_i = \sum W_{ij}S_j$ and the state update are done in parallel optoelectronically at considerable advantage in iteration speed over purely electronic systems.

Progress in amorphous silicon liquid crystal spatial light modulators (a:s_i LCSLMs) is providing sensitive optically addressable nonvolatile devices of (3x3)cm^2 active area with better than 100 ℓp/mm resolution, .03 pJ addressing energy per pixel and a speed of over 10^3 frames/sec. [4]. Moreover the device is nonvolatile. These mean that such a device can furnish ~10^{10} modifiable synaptic weights per second provided that an optical means for downloading these weights from a computer controller into the device via a CRT display at the high rate of 10 G bits/sec is found. Wide-band microchannel-plate assisted CRTs of the variety used by Tecktronix in their GHz bandwidth oscilloscopes or multiple electron beam CRTs can be considered for this task.

Pixelized a:s; LCSLM are also under construction with similar resolution but smaller number of pixels, e.g. 128x128 pixels at present with 256x256 and larger arrays being under development [5]. These arrays are addressed electronically and act as an optically interrogatable static RAM.

The above line of reasoning moves us to conclude that as the development of spatial light modulators proceeds towards larger sizes and faster frame rate devices that are nonvolatile and ı ˙. 4-5 bits of pixel dynamic range, the photonic approach to constructing versatile neurocomputers can offer distinct advantages over the purely electronic approach.

Finally our work in neuromorphic target identification indicates that greater attention should be given to the issue of cognition in neural networks. Our preliminary findings indicate that networks which compute with periodic attractors, instead of fixed point attractors, have interesting capabilities and that studying bifurcating networks may offer a possible mechanism for cognition.

4. LIST OF PUBLICATIONS

- N.H. Farhat, "Optoelectronic Neural Networks and Learning Machines", IEEE Circuits and Devices Magazine, Vol. 5, pp. 32-41, Sept. 1989 (Invited).
- N. Farhat and Z.Y. Shae, "An Optical Learning Machine", Proc. 1989 Hawaii International Conference on Systems Science, Vol. 1, IEEE Computer Society Press, IEEE Cat. No. 89TH0242-8, p. 432-439, 1989.

- N.H. Farhat and H. Babri, "Phase-Space Engineering for Neurodynamic Target Identification", Proc. IEEE-APS Int. Symp. and URSI Radio Science Meeting, Vol. II, pp. 768-771, IEEE Cat. No. CH-2654-2/89, June 1989.

5. REFERENCES

1. W.J. Freeman AND G. Viana Di Prisco, "EEG Spatial Pattern Differences with Discriminated Odors Manifest Chaotic and Limit Cycle Attractors in Olfactory Bulb of Rabbits," Prc. Conf. on Brain Theory, Trieste 1984, Springer Verlag, Berlin, Heidelberg, New York, Tokyo, (1984).

2. D. Psaltis and N. Farhat, "Optical Information Processing Based on Associative Memory Model of Neural Nets with Thresholding and Feedback," Optics Letters, Vol. 10, p. 98 (1985).

3. N. Farhat et. al., "Optical Implementation of the Hopfield Model," App. Opt., Vol. 24, pp. 1469-2475, May 1985.

4. S. Yamamoto, et. al., "Optical Pattern Recognition with LAPS-SLM(1) (Light Addressed Photoconductor and Smectic C* Liquid Crystal Spatial Light Modulator," SPIE Meeting, Los Angeles, 1989.

5. Applied Optics, Special Issue on Spatial Light Modulators for Optical Information Processing, November 1989.

6. APPENDICES

I. Optoelectronic Neural Networks and Learning Machines.

II. Stochastic Optical Learning Machine

III. Learning Networks for Extrapolation and Radar Target Identification

IV. Phase Space Engineering for Neurodynamic Target Identification

Optoelectronic Neural Networks and Learning Machines

Nabil H. Farhat

Foreword

Circuits and Devices Magazine is featuring three sequential articles on the current status of artificial neural network implementation technology. The current offering, on optronic implementation of artificial neural networks, is the second entry in this trilogy. It is sandwiched between the previous overview on analog implementation and the upcoming survey of digital artificial neural networks.

Nabil H. Farhat, who penned this overview, is a co-author of the 1985 article in Optics Letters and follow-up paper in Applied Optics that broke ground for modern optical implementation of artificial neural networks.

Robert J. Marks II

Abstract

Optics offers advantages in realizing the parallelism, massive interconnectivity, and plasticity required in the design and construction of large-scale optoelectronic (photonic) neurocomputers that solve optimization problems at potentially very high speeds by learning to perform mappings and associations. To elucidate these advantages, a brief neural net primer based on phase-space and energy landscape considerations is first presented. This provides the basis for subsequent discussion of optoelectronic architectures and implementations with self-organization and learning ability that are configured around an optical crossbar interconnect. Stochastic learning in the context of a Boltzmann machine is then described to illustrate the flexibility of optoelectronics in performing tasks that may be difficult for electronics alone. Stochastic nets are studied to gain insight into the possible role of noise in biological neural nets. We close by describing two approaches to realizing large-scale optoelectronic neurocomputers: integrated optoelectronic neural chips with interchip optical interconnects that enables their clustering into large neural networks, and nets with two-dimensional rather than one-dimensional arrangement of neurons and four-dimensional connectivity matrices for increased packing density and compatibility with two-dimensional data. We foresee integrated optoelectronics or photonics playing an increasing role in the construction of a new generation of versatile programmable analog computers that perform computations collectively for use in neuromorphic (brain-like) processing and fast simulation and study of complex nonlinear dynamical systems.

Introduction

Neural net models and their analogs offer a brain-like approach to information processing and representation that is distributed, nonlinear and iterative. Therefore they are best described in terms of phase-space behavior where one can draw upon a rich background of theoretical results developed in the field of nonlinear dynamical systems. The ultimate purpose of biological neural nets (BNNs) is to sustain and enhance survivability of the organism they reside in, doing so in an imprecise and usually very complex environment where sensory impressions are at best sketchy and difficult to make sense of had they been treated and analyzed by conventional means. Embedding artificial neural nets (ANNs) in man-made systems endows them therefore with enhanced survivability through fault-tolerance, robustness and speed. Furthermore, survivability implies adaptability through self-organization, knowledge accumulation and learning. It also implies lethality.

All of these are concepts found at play in a wide range of disciplines such as economics, social science, and even military science which can perhaps explain the widespread interest in neural nets exhibited today from both intellectual and technological viewpoints. It is widely believed that artificial neurocomputing and knowledge processing systems could eventually have significant impact on information processing, pattern recognition, and control. However, to realize the potential advantages of neuromorphic processing, one must contend with the issue of how to carry out collective neural computation algorithms at speeds far beyond those possible with digital computing. Obviously parallelism and concurrency are essential ingredients and one must contend with basic implementation issues of how to achieve such massive connectivity and parallelism and how to achieve artificial plasticity, i.e., adaptive modification of the strength of interconnections (synaptic weights) between neurons that is needed for memory and self-programming (self-organization and learning). The answers to these questions seem to be coming from two directions of research. One is connection machines in which a large number of digital central processing units are interconnected to perform parallel computations in VLSI hardware; the other is analog hardware where a large number of simple processing units (neurons) are connected through modifiable weights such that their phase-space dynamic behavior has useful signal processing functions associated with it.

Analog optoelectronic hardware implementation of neural nets (see Farhat et al. in list of further reading), since first introduced in 1985, has been the focus of attention for several reasons. Primary among these is that the optoelectronic or photonic approach combines the best of two worlds: the massive interconnectivity and parallelism of optics and the flexibility, high gain, and decision making capability (nonlinearity) offered by electronics. Ultimately, it seems more attractive to form analog neural hardware by completely optical means where switching of signals from optical to electronic carriers and vice versa is avoided. However, in the absence of suitable fully optical decision making devices (e.g., sensitive optical bistability devices), the capabilities

8755-3996/89/0900-0032$1.00 © 1989 IEEE

Vol. 5 No. 5

of the optoelectronic approach remain quite attractive and could in fact remain competitive with other approaches when one considers the flexibility of architectures possible with it.* In this paper we concentrate therefore on the optoelectronic approach and give selected examples of possible architectures, methodologies and capabilities aimed at providing an appreciation of its potential in building a new generation of programmable analog computers suitable for the study of non-linear dynamical systems and the implementation of mappings, associative memory, learning, and optimization functions at potentially very high speed.

We begin with a brief neural net primer that emphasizes phase-space description, then focus attention on the role of optoelectronics in achieving massive interconnectivity and plasticity. Architectures, methodologies, and suitable technologies for realizing optoelectronic neural nets based on optical crossbar (matrix vector multiplier) configurations for associative memory function are then discussed. Next, partitioning an optoelectronic analog of a neural net into distinct layers with a prescribed interconnectivity pattern as a prerequisite for self-organization and learning is discussed. Here the emphasis will be on stochastic learning by simulated annealing in a Boltzmann machine. Stochastic learning is of interest because of its relevance to the role of noise in biological neural nets and because it provides an example of a task that demonstrates the versatility of optics. We close by describing several approaches to realizing the large-scale networks that would be required in analog solution of practical problems.

Neural Nets—A Brief Overview

In this section, a brief qualitative description of neural net properties is given. The emphasis is on energy landscape and phase-space representations and behavior. The descriptive approach adopted is judged best as background for appreciating the material in subsequent sections without having to get involved in elaborate mathematical exposition. All neural net properties described here are well known and can easily be found in the literature. The viewpoint of relating all neural net properties to energy landscape and phase-space behavior is also important and useful in their classification.

A neural net of N neurons has (N^2-N) interconnections or $(N^2-N)/2$ symmetric interconnections, assuming that a neuron does not communicate with itself. The state of a neuron in the net, i.e., its firing rate, can be taken to be binary (0, 1) (on-off, firing or not firing) or smoothly varying according to a nonlinear continuous monotonic function often taken as a sigmoidal function bounded from above and below. Thus the state of the i-th neuron in the net can be described mathematically by

$$s_i = f\{u_i\} \qquad i = 1, 2, 3 \ldots N^{**} \tag{1}$$

where $f\{.\}$ is a sigmoidal function and

$$u_i = \sum_{j=1}^{N} W_{ij}s_j - \theta_i + I_i \tag{2}$$

is the activation potential of the i-th neuron, W_{ij} is the strength or weight of the synaptic interconnection between the j-th neuron and the i-th neuron, and $W_{ii}=0$ (i.e., neurons do not talk to themselves). θ_i and I_i are, respectively, the threshold level and external or control input to the i-th neuron, thus $W_{ij}S_j$ represents the input to neuron i from neuron j and the first term on the right side of (2) represents the sum of all such inputs to the i-th neuron. For excitatory interconnections or synapses, W_{ij} is positive, and it is negative for inhibitory ones. For a binary neural net, that is, one in which the nurons are binary, i.e., $s_i[0,1]$, the smoothly varying function $f\{.\}$ is replaced by $U\{.\}$, where U is the unit step function. When W_{ij} is symmetric, i.e., $W_{ij}=W_{ji}$, one can define (see J. J. Hopfield's article in list of further reading) a Hamiltonian or energy function E for the net by

$$E = -\frac{1}{2}\sum_i u_i s_i$$

$$= -\frac{1}{2}\sum_i\sum_j W_{ij}s_i s_j - \frac{1}{2}\sum_i (\theta_i - I_i)s_i \tag{3}$$

The energy is thus determined by the connectivity matrix W_{ij}, the threshold level θ_i and the external input I_i. For symmetric W_{ij} the net is stable; that is, for any threshold level θ_i and given "strobed" (momentarily applied) input I_i, the energy of the net will be a decreasing function of the neurons state s_i of the net or a constant. This means that the net always heads to a steady state of local or global energy minimum. The descent to an energy minimum takes place by the iterative discrete dynamical process described by Eqs. (1) and (2) regardless of whether the state update of the neurons is synchronous or asynchronous. The minimum can be local or global, as the "energy landscape" of a net (a visualization of E for every state s_i) is not monotonic but will possess many uneven hills and troughs and is therefore characterized by many local minima of various depths and one global (deepest) minimum. The energy landscape can therefore be modified in accordance with Eq. (3) by changing the interconnection weights W_{ij} and/or the threshold levels θ_i and/or the external input I_i. This ability to "sculpt" the energy landscape of the net provides for almost all the rich and fascinating behavior of neural nets and for the ongoing efforts of harnessing these properties to perform sophisticated spatio-temporal mappings, computations, and control functions. Recipes exist that show how to compute the W_{ij} matrix to make the local energy minima correspond to specific desired states of the network. As the energy minima are stable states, the net tends to settle in one of them, depending on the initializing state, when strobed by a given input. For example, a binary net of N=3 neurons will have a total of $2^N=8$ states. These are listed in Table 1. They represent all possible combinations s_1, s_2 and s_3 of the three neurons that describe the state vector $s=[s_1,s_2,s_3]$ of the net. For a net of N neurons the state vector is N-dimensional. For N=3 the state vector can be represented as a point (tip of a position vector) in 3-D

*It is worth mentioning here that recent results obtained in our work show that networks of logistic neurons, whose response resembles that of the derivative of a sigmoidal function, exhibit rich and interesting dynamics, including spurious state-free associative recall, and allow the use of unipolar synaptic weights. The networks can be realized in a large number of neurons when implemented with optically addressed reflection-type liquid crystal spatial light modulators. However, the flexibility of such an approach versus that of the photonic approach is yet to be determined.

**From here on it will be taken as understood that whenever the subscripts (i or j) appear, they run from 1 up to N where N is the number of neurons in the net.

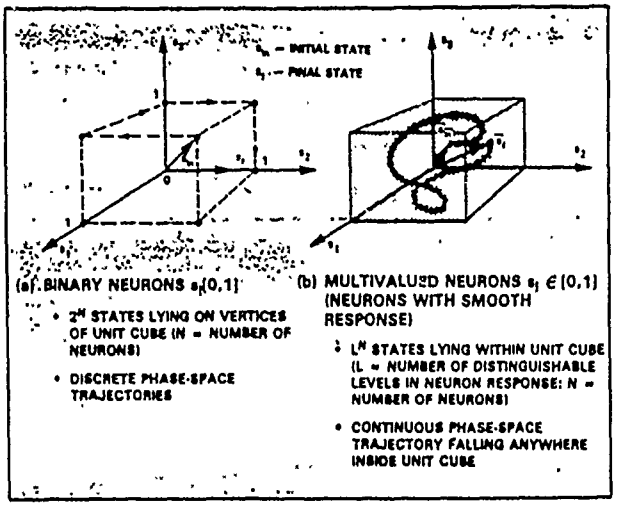


Fig. 1 Phase-space or state space representation and trajectories for a neural net of N = 3 neurons. (a) for binary neurons, (b) for neurons with normalized smooth (sigmoidal) response.

space. The eight state vectors listed in Table I fall then on the vertices of a unit cube as illustrated in Fig. 1(a). As the net changes its state, the tip of the state vector jumps from vertex to vertex describing a discrete trajectory as depicted by the broken trajectory starting from the tip of the initializing state vector s_i and ending at the tip of the final state vector s_f. For any symmetric connectivity matrix assumed for the three-neuron net example, each of the eight states in Table I yields a value of the energy E. A listing of these values for each state represents the energy landscape of the net.

For a nonbinary neural net whose neurons have normalized sigmoidal response $s_i \in [0,1]$, i.e., s_i varies smoothly between zero and one, the phase-space trajectory is continuous and is always contained within the unit cube as illustrated in Fig. 1(b). The neural net is governed then by a set of continuous differential equations rather than the discrete update relations of Eqs. (1) and (2). Thus one can talk of nets with either discrete or continuous dynamics. The above phase-space representation is extendable to a neural net of N neurons where one considers discrete trajectories between the vertices of a unit hypercube in N-dimensional space or a smooth trajectory confined within the unit hypercube for discrete and continuous neural nets, respectively.

The stable states of the net, described before as minima of the energy landscape, correspond to points in the phase-space towards which the state of the net tends to evolve in time when the net is iterated from an arbitrary initial state. Such stable points are called "attractors" or "limit points" of the net, to borrow from terms used in the description of nonlinear dynamical systems. Attractors in phase-space are characterized by basins of attraction of given size and shape. Initializing the net from a state falling within the basin of attraction of a given attractor and thus regarded as an incomplete or noisy version of the attractor, leads to a trajectory that converges to that attractor. This is a many to one mapping or an associative search operation that leads to an associative memory attribute of neural nets.

Table I. Possible States of a Binary Neural Net of 3 Neurons

s_1	s_2	s_3
0	0	0
0	0	1
0	1	0
1	0	0
0	0	1
1	0	1
1	1	0
1	1	1

Local minima in an energy landscape or attractors in phase-space can be fixed by forming W_{ij} in accordance with the Hebbian learning rule (see both Hebb and Hopfield in list of further reading), i.e., by taking the sum of the outer products of the bipolar versions of the state vector we wish to store in the net

$$W_{ij} = \sum_{m=1}^{M} v_i^{(m)} v_j^{(m)} \tag{4}$$

where

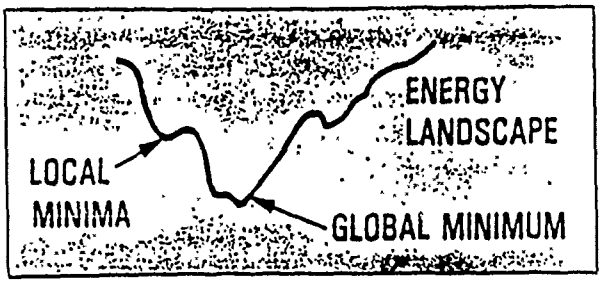


Fig. 2 Conceptual representation of energy landscape.

$$v_i^{(m)} = 2s_i^{(m)} - 1 \qquad i = 1,2\ldots N \qquad m = 1,2\ldots M \tag{5}$$

are M bipolar binary N-vectors we wish to store in the net. Provided that $s_i^{(m)}$ are uncorrelated and

$$M \leq \frac{N}{4\ell n N} \tag{6}$$

the M stored state $s^{(m)}$ will become attractors in phase-space of the net or equivalently their associated energies will be local minima in the energy landscape of the net as illustrated conceptually in Fig. 2. As M increases beyond the value given by (6), the memory is overloaded, spurious local minima are created in addition to the desired ones and the probability of correct recall from partial or noisy information deteriorates, compromising operation of the net as an associative memory (see R.J. McEliece et al. in list of further reading).

The net can also be formed in such a way as to lead to a hetero-associative storage and recall function by setting the interconnection weights in accordance with

$$W_{ij} = \sum_{m} v_i^{(m)} g_j^{(m)} \tag{7}$$

where $\bar{v}^{(m)}$ and $\bar{g}^{(m)}$ are associated N-vectors. Networks of this variety can be used as feedforward networks only and this precludes the rich dynamics encountered in feedback or recurrent networks from being observed. Nevertheless, they are useful for simple mapping and representation.

Energy landscape considerations are useful in devising formulas for the storage of sequences of associations or a cyclic sequence of associations as would be required for conducting sequential or cyclic searches of memories.

Learning in biological neural nets is thought to occur by self-organization where the synaptic weights are modified electrochemically as a result of environmental (sensory and other (e.g., contextual)) inputs. All such learning requires plasticity, the process of gradual synaptic modification. Adaptive learning algorithms can be deterministic or stochastic; supervised or unsupervised. An optoelectronic (Boltzmann machine) and its learning performance will be described in the section on large scale networks as an illustration of the unique capabilities of optoelectronic hardware.

Neural Nets Classification and Useful Functions

The energy function and energy landscape description of the behavior of neural networks presented in the preceding sections allows their classification into three groups. For one group the local minima in the energy landscape are what counts in the network's operation. In the second group the local minima are not utilized and only the global minimum is meaningful. In the third group the operations involved do not require energy considerations. They are merely used for mapping and reduction of dimensionality. The first group includes Hopfield-type nets for all types of associative memory applications that include auto-associative, hetero-associative, sequential and cyclic data storage and recall. This category also includes all self-organizing and learning networks regardless of whether the learning in them is supervised, unsupervised, deterministic, or stochastic as the ultimate result of the fact that learning, whether hard or soft, can be interpreted as shaping the energy landscape of the net so as to "dig" in it valleys corresponding to learned states of the network. All nets in this category are capable of generalization. An input that was not learned specifically but is within a prescribed Hamming distance* to one of the entities learned would elicit, in the absence of any contradictory information, an output that is close to the outputs evoked when the learned entity is applied to the net. Because of the multilayered and partially interconnected nature of self-organizing networks, one can define input and output groups of neurons that can be of unequal number (See section on large scale networks). This is in contrast to Hopfield-type nets which are fully interconnected and therefore the number of input and output neurons is the same (the same neurons define the initial and final states of the net). The ability to define input and output groups of neurons in multilayered nets enables additional capabilities that include learning, coding, mapping, and reduction of dimensionality.

The second group of neural nets includes nets that perform calculations that require finding the global energy minimum of the net. The need for this type of calculation often occurs in combinatorial optimization problems and in the solution of inverse problems encountered, for example, in vision, remote sensing, and control.

The third group of neural nets is multilayered with localized nonglobal connections similar to those in cellular automata where each neuron communicates within its layer with a pattern of neurons in its neighborhood and with a pattern of neurons in the next adjacent layer. Multilayered nets with such localized connections can be used for mapping and feature extraction. Neural nets can also be categorized by whether they are single layered or multilayered, self-organizing or nonself-organizing, solely feedforward or involve feedback, stochastic or deterministic. However, the most general categorization appears to be in terms of the way the energy landscape is utilized, or in terms of the kind of attractors formed and utilized in its phase-space (limit points, limit cycles, or chaotic**).

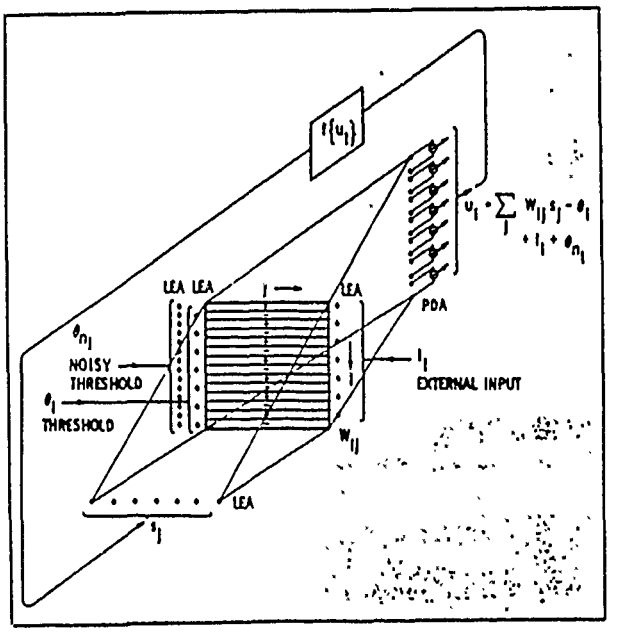


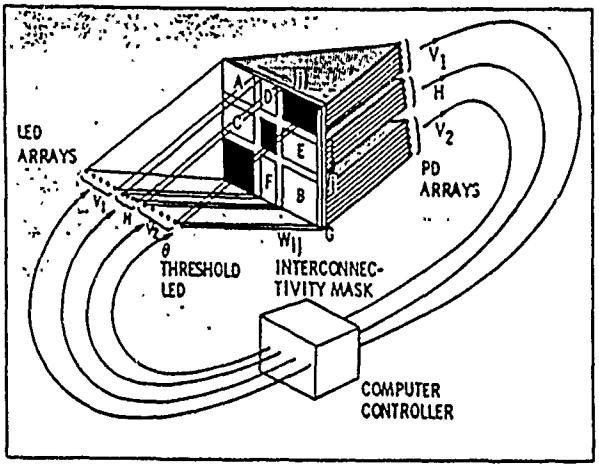
Fig. 3 Optoelectronic analog circuit of a fully interconnected neural net.

Implementations

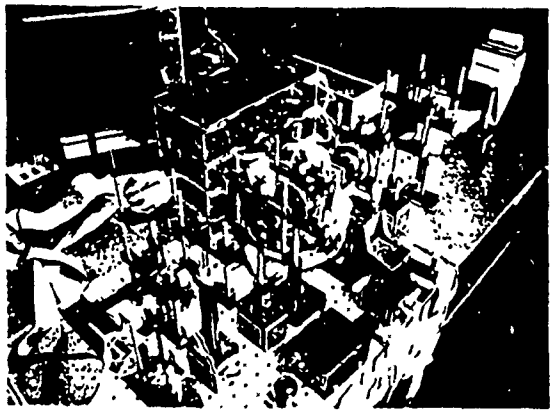
The earliest optoelectronic neurocomputer was of the fully interconnected variety where all neurons could talk to each other. It made use of incoherent light to avoid interference effects and speckle noise and also relax the stringent alignment required in coherent light systems. An optical crossbar interconnect (see Fig. 3) was employed to carry out the vector matrix multiplication operation required in the summation term in Eq. 2. (see Farhat et al. (1985) in list of further reading). In this arrangement the state vector of the neurons is represented by the linear light emitting array (LEA) or equivalently by a linear array of light modulating elements of a spatial light modulator (SLM), the connectivity matrix W_{ij} is implemented in a photographic transparency mask (or a 2-D SLM when a modifiable connectivity mask is needed for adaptive learning), and the activation potential u_i is measured with a photodiode array (PDA). Light from the LEA is smeared vertically onto the W_{ij} mask with

*The Hamming distance between two binary N-vectors is the number of elements in which they differ.

**A chaotic attractor is manifested by a phase-space trajectory that is completely unpredictable and is highly sensitive to initial conditions. It could ultimately turn out to play a role in cognition.



(a)



(b)

Fig. 4 Boltzmann learning machine. (a) optoelectronic circuit diagram of a net partitioned into three layers by blocking segments of the interconnectivity mask, (b) hardware implementation showing the state vector LED array at the top right, the MOSLM at the center (between lenses) and an intensified PDA (PDA abutted to an image intensifier fiber output window for added gain) in the lower left. The integrated circuit board rack contains the MOSLM driver and computer interface and the TV receiver in the background provides the "snow pattern" that is imaged through a slit onto the intensifier input window for optical injection of noise in the network.

the aid of an anamorphic lens system (cylindrical and spherical lenses in tandem not shown in the figure for simplicity). Light passing through rows of W_{ij} is focused onto the PDA elements by another anamorphic lens system. To realize bipolar transmission values in incoherent light, positive elements and negative elements of any row of W_{ij} are assigned to two separate subrows of the mask and light passing through each subrow is focused onto adjacent pairs of photosites of the PDA whose outputs are subtracted. In Fig. 3, both the neuron threshold θ_i and external input I_i are injected optically with the aid of a pair of LEAs whose light is focused on the PDA. Note that positive valued I_i is assumed here and therefore its LEA elements are shown positioned to focus onto positive photosites of the PDA only.

This architecture was successfully employed in the first implementation of a 32 neuron net (see Farhat et al. (1985) in list of further reading). Fig. 3 also shows a third LEA for injection of spatio-temporal noise into the net as would be required, for example, in the implementation of a noisy threshold scheme for the Boltzmann learning machine to be discussed later. The net of Fig. 3 behaved as an associative memory very much as expected and was found to exhibit correct recovery of three neurons stored from partial information and showed robustness with element failure (two of its 32 neurons were accidentally disabled, 2 PDA elements broke, and no noticeable degradation in performance was observed).

In the arrangement of Fig. 3, the neurons are fully interconnected. To implement learning in a neural net, one needs to impart structure to the net, i.e., be able to partition the net into distinct input, output, and hidden groups or layers of neurons with a prescribed pattern of communication or interconnections between them which is not possible in a fully interconnected or single layer network. A simple but effective way of partitioning a fully interconnected optoelectronic net into several layers to form a partially interconnected net is shown in Fig. 4(a). This is done simply by blocking certain portions of the W_{ij} matrix.

In the example shown, the blocked submatrices serve to prevent neurons from the input group V_1 and the output group V_2 from talking to each other directly. They can do so only via the hidden or buffer group of neurons H. Furthermore, neurons within H can not talk to each other. This partition scheme enables arbitrary division of neurons among layers and can be rapidly set when a programmable nonvolatile SLM under computer control is used to implement the connectivity weights. Neurons in the input and output groups are called visible neurons because they interface with the environment.

The architecture of Fig. 4 can be used in supervised learning where, beginning from an arbitrary W_{ij}, the net is presented with an input vector from the training set of vectors it is required to learn through V_1 and its convergent output state is observed on V_2 and compared with the desired output (association) to produce an error signal which is used in turn according to a prescribed formula to update the weights matrix. This process of error-driven adaptive weights modification is repeated a sufficient number of times for each vector and all vectors of the training set until inputs evoke the correct desired output or association at the output. At that time the net can be declared as having captured the underlying structure of the environment (the vectors presented to it) by forming an internal representation of the rules governing the mappings of inputs into the required output associations.

Many error-driven learning algorithms have been proposed and studied. The most widely used, the error back-projection algorithm (see Werbos, Parker, and Rumelhart et al. in list of further reading), is suited for use in feed forward multilayered nets that are void of feedback between the neurons. The architecture of Fig. 4(a) has been successfully employed in the initial demonstration of supervised stochastic learning by simulated annealing. Our interest in stochastic learning stemmed from a desire to better understand the possible role of noise in BNNs and to find means for accelerating the simulated annealing process through the use of optics and optoelectronic hardware. For any input-output association clamped on V_1 and V_2 and beginning from an arbitrary W_{ij} that could be random, the net is annealed through the hidden neurons by subjecting them to optically injected noise in the form of a

noise component added to the threshold values of the neurons as depicted by θ_{ni} in Fig. 3.

The source of controlled noise used in this implementation was realized by imaging a slice of the familiar "snow pattern" displayed on an empty channel of a television receiver, whose brightness could be varied under computer control, onto the PD array of Fig. 4(a). This produces controlled perturbation or "shaking" of the energy landscape of the net which prevents its getting trapped into a state of local energy minimum during iteration and guarantees its reaching and staying in the state of the global energy minimum or one close to it. This requires that the injected noise intensity be reduced gradually, reaching zero when the state of global energy minimum is reached to ensure that the net will stay in that state. Gradual reduction of noise intensity during this process is equivalent to reducing the "temperature" of the net and is analogous to the annealing of a crystal melt to arrive at a good crystalline structure. It has accordingly been called simulated annealing by early workers in the field.

Finding the global minimum of a "cost" or energy function is a basic operation encountered in the solution of optimization problems and is found not only in stochastic learning. Mapping optimization problems into stochastic nets of this type, combined with fast annealing to find the state of global "cost function" minimum, could be a powerful tool for their solution. The net behaves then as a stochastic dynamical analog computer. In the case considered here, however, optimization through simulated annealing is utilized to obtain and list the convergent states at the end of annealing bursts when the training set of vectors (the desired associations) are clamped to V_1 and V_2. This yields a table or listing of convergent state vectors from which a probability P_{ij} of finding the i-th neuron and the j-th neuron on at the same time is computed. This completes the first phase of learning. The second phase of learning involves clamping the V_1 neurons only and annealing the net through H and V_2, obtaining thereby another list of convergent state vectors at the end of annealing bursts and calculating another probability P'_{ij} of finding the i-th and j-th neurons on at the same time. The connectivity matrix, implemented in a programmable magneto-optic SLM (MOSLM), is modified then by $\Delta W_{ij} = \epsilon(P_{ij} - P'_{ij})$ computed by the computer controller where ϵ is a constant controlling the learning rate. This completes one learning cycle or episode. The above process is repeated again and again until the W_{ij} stabilizes and captures hopefully the underlying structure of the training set. Many learning cycles are required and the learning process can be time-consuming unless the annealing process is sufficiently fast.

We have found that the noisy thresholding scheme leads the net to anneal and find the global energy minimum or one close to it in about 35 time constants of the neurons used. For microsecond neurons this could be 10^4-10^5 times faster than numerical simulation of stochastic learning by simulated annealing which requires random selection of neurons one at a time, switching their states, and accepting the change of state in such a way that changes leading to an energy decrease are accepted and those leading to energy increases are allowed with a certain controlled probability.

The computer controller in Fig. 4 performs several functions. It clamps the input/output neurons to the desired states during the two phases of learning, controls the annealing profile during annealing bursts, monitors the convergent state vectors of the net, and computes and executes the weights modification. For reasons related to the thermodynamical and statistical mechanical interpretation of its operation, the architecture in Fig. 4(a) is called a Boltzmann learning machine. A pictorial view of an optoelectronic (photonic) hardware implementation of a fully operational Boltzmann learning machine is shown in Fig. 4(b). This machine was built around a MOSLM as the adaptive weights mask.

The interconnection matrix update during learning requires small analog modifications ΔW_{ij} in W_{ij} Pixel transmittance in the MOSLM is binary, however. Therefore a scheme for learning with binary weights was developed and used in which W_{ij} is made 1 if $(P_{ij} - P'_{ij}) > M$ regardless of its preceeding value, where M is a constant, and made -1 if $(P_{ij} - P'_{ij}) < -M$ regardless of its preceeding value, and is left unchanged if $-M \geq (P_{ij} - P'_{ij}) \leq M$. This introduces inertia to weights modification and was found to allow a net of N = 24 neuron partitioned into 8-8-8 groups to learn two autoassociations with 95 percent score (probability of correct recall) when the value of M was chosen randomly between (0-.5) for each learning cycle. This score dropped to 70 percent in learning three autoassociations. However, increasing the number of hidden neurons from 8 to 16 was found to yield perfect learning (100 percent score).

Scores were collected after 100 learning cycles by computing probabilities of correct recall of the training set. Fast annealing by the noisy thresholding scheme was found to scale well with size of the net, establishing the viability of constructing larger optoelectronic learning machines. In the following section two schemes for realizing large-scale nets are briefly described. One obvious approach discussed is the clustering of neural modules or chips. This approach requires that neurons in different modules be able to communicate with each other in parallel, if fast simulated annealing by noisy thresholding is to be carried out. This requirement appears to limit the number of neurons per module to the number of interconnects that can be made from it to other modules. This is a thorny issue in VLSI implementation of cascadeable neural chips (see Alspector and Allen in list of further reading). It provides a strong argument in favor of optoelectronic neural modules that have no such limitation because communication between modules is carried out by optical means and not by wire.

Large Scale Networks

To date most optoelectronic implementations of neural networks have been prototype units limited to few tens or hundreds of neurons. Use of neurocomputers in practical applications involving fast learning or solution of optimization problems requires larger nets. An important issue, therefore, is how to construct larger nets with the programmability and flexibility exhibited by the Boltzmann learning machine prototype described. In this section we present two possible approaches to forming large-scale nets as examples demonstrating the viability of the photonic approach. One is based on the concept of a clusterable integrated optoelectronic neural chip or module that can be optically interconnected to form a larger net, and the second is an architecture in which 2-D arrangement of neurons is utilized, instead of the 1-D arrangement described

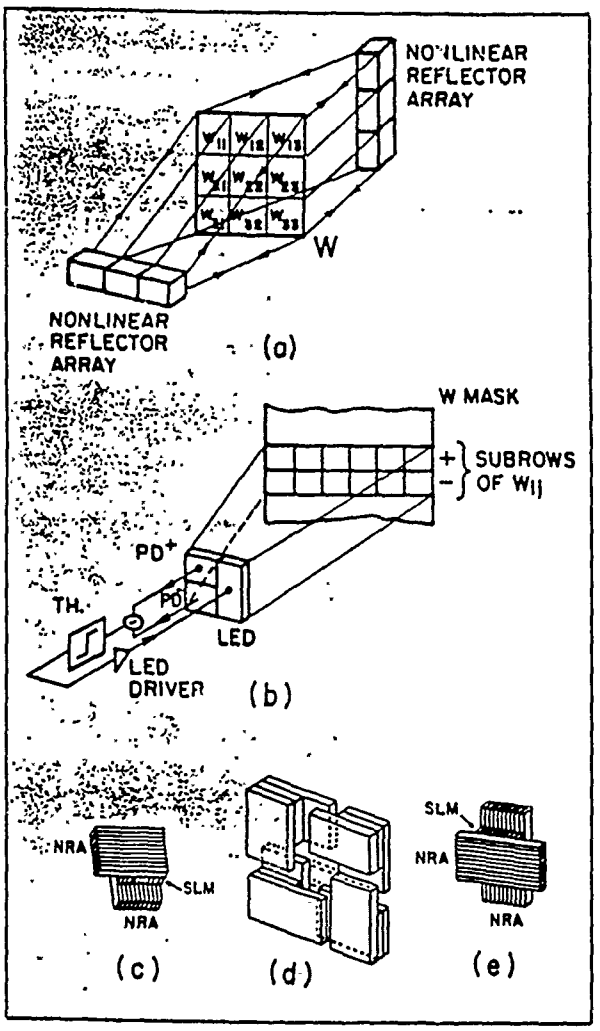


Fig. 5 Optoelectronic neural net employing internal feedback and two orthogonal nonlinear reflector arrays (NRAs) consisting of channels of nonlinear light amplifiers (photodetectors, thresholding amplifiers, LEDs and LED drivers). a) architecture, (b) detail of mask and single element of nonlinear reflector array, (c) and (d) optoelectronic neural chip concept and cluster of four chips, (e) neural chip for forming clusters of more than four chips.

in earlier sections, in order to increase packing density and to provide compatibility with 2-D sensory data formats.

Clusterable Photonic Neural Chips

The concept of a clusterable photonic neural chip, which is being patented by the University of Pennsylvania, is arrived at by noting that when the connectivity matrix is symmetrical, the architectures we described earlier (see Figs. 3 or 4(a)) can be modified to include internal optical feedback and nonlinear "reflection" (optoelectronic detection, amplification, thresholding and light emission or modulation) on both sides of the connectivity mask W or nonvolatile SLM (e.g., a MOSLM) as depicted in Fig. 5 (see Farhat (1987) in list of further reading). The nonlinear reflector arrays are basically retro-reflecting optoelectronic or photonic light amplifier arrays that receive and retransmit light on the same side facing the MOSLM.

Two further modifications are needed to arrive at the concept of clusterable integrated optoelectronics or photonic neural chips. One is replacement of the LEDs of the nonlinear reflector arrays by suitable spatial light modulators of the fast ferroelectric liquid crystal variety for example, and extending the elements of the nonlinear reflector arrays to form stripes that extend beyond the dimensions of the connectivity SLM, and sandwiching the latter between two such striped nonlinear reflector arrays oriented orthogonally to each other as depicted in Fig. 5(c). This produces a photonic neural chip that operates in an ambient light environment. Analog integrated circuit (IC) technology would then be used to fabricate channels of nonlinear (thresholding) amplifiers and SLM drivers, one channel for each PD element. The minute IC chip thus fabricated is mounted as an integral part on each PDA/SLM assembly of the nonlinear reflector arrays. Individual channels of the IC chip are bonded to the PDA and SLM elements. Two such analog IC chips are needed per neural chip. The size of the neural chip is determined by the number of pixels in the SLM used.

An example of four such neural chips connected optoelectronically to form a larger net by clustering is shown in Fig. 5(d). This is achieved by simply aligning the ends of the stripe PD elements in one chip with the ends of the stripe SLM elements in the other. It is clear that the hybrid photonic approach to forming the neural chip would ultimately and preferably be replaced by an entirely integrated photonic approach and that neural chips with the slightly different form shown in Fig. 5(e) can be utilized to form clusters of more than four. Large-scale neural nets produced by clustering integrated photonic neural chips have the advantage of enabling any partitioning arrangement, allowing neurons in the partitioned net to communicate with each other in the desired fashion enabling fast annealing by noisy thresholding to be carried out, and of being able to accept both optically injected signals (through the PDAs) or electronically injected signals (through the SLMs) in the nonlinear reflector arrays, facilitating communication with the environment. Such nets are therefore capable of both deterministic or stochastic learning. Computer controlled electronic partitioning and loading and updating of the connectivity weights in the connectivity SLM (which can be of the magneto-optic variety or the nonvolatile ferroelectric liquid crystal (FeLCSLM) variety) is assumed. This approach to realizing large-scale fully programmable neural nets is currently being developed in our laboratory, and illustrates the potential role integrated photonics could play in the design and construction of a new generation of analog computers intended for use in neurocomputing and rapid simulation and study of nonlinear dynamical systems.

Neural Nets with Two-Dimensional Deployment of Neurons

Neural net architectures in which neurons are arranged in a two-dimensional (2-D) format to increase packing density and to facilitate handling 2-D formatted data have received early attention (see Farhat and Psaltis (1987) in list of further reading). These arrangements involve a 2-D N × N state "vector" or matrix s_{ij} representing the state of neurons, and a four-dimensional (4-D) connectivity "matrix" or tensor T_{ijkl} representing the weights of synapses between neurons. A scheme for partitioning the 4-D connectivity tensor into an N × N array of submatrices, each

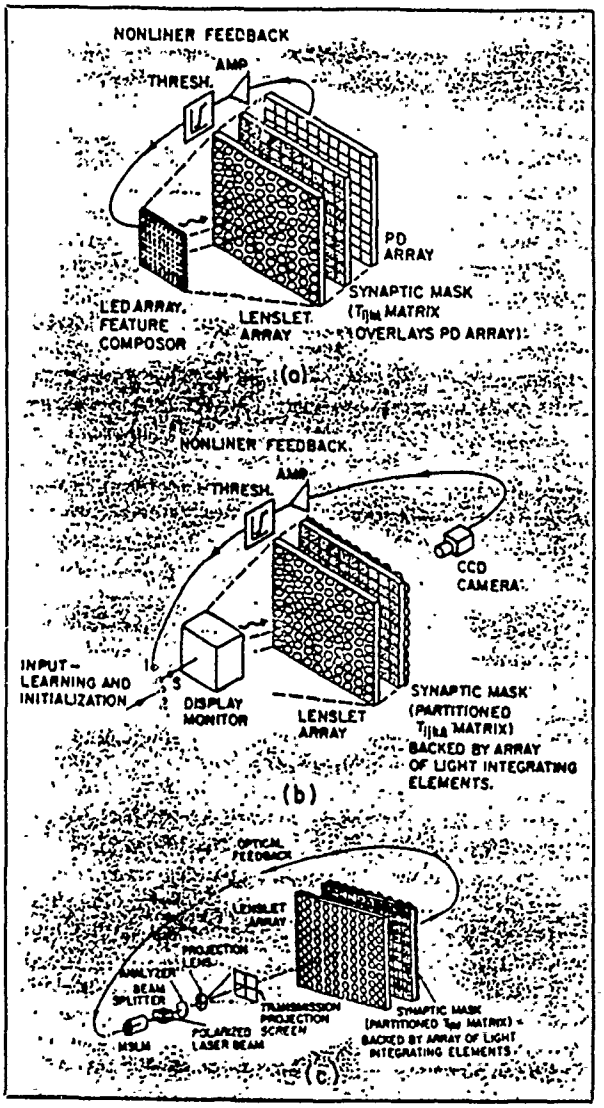


Fig. 6 Three optoelectronic network architectures in which the neurons are arranged in two-dimensional format employing: (a) parallel nonlinear electronic amplification and feedback, (b) serial nonlinear electronic amplification and feedback, (c) parallel nonlinear electron optical amplification and feedback.

of which has N × N elements, to enable storing it in a flat 2-D photomask or SLM for use in optoelectronic implementation has been developed (see Farhat and Psaltis 1987 in list of further reading). Several arrangements are possible using this partitioning scheme (see Fig. 6).

In Fig. 6(a), neuron states are represented with a 2-D LED array (or equivalently with a 2-D SLM). A two-dimensional lenslet array is used to spatially multiplex and project the state vector display onto each of the submatrices of the partitioned connectivity mask. The product of the state matrix with each of the weights stored in each submatrix is formed with the help of a spatially integrating square photodetector of suitable size positioned behind each submatrix. The (i-j)th photodetector output represents the activation potentials u_{ij} of the (i-j)th neurons. These activation potentials are nonlinearly amplified and fed back in parallel to drive the corresponding elements of the LED state array of those of the state SLM. In this fashion, weighted interconnections between all neurons are established by means of the lenslet array instead of the optical crossbar arrangement used to establish connectivity between neurons when they are deployed on a line.

Both plastic molded and glass micro-lenslet arrays can be fabricated today in 2-D formats. Glass micro-lenslet arrays with density of 9 to 25 lenslets/mm^2 can be made in large areas using basically photolithographic techniques. Resolution of up to ~50 ℓp/mm can also be achieved. Therefore, a micro lenslet array of (100 × 100)mm^2, for example, containing easily 10^5 lenslets could be used to form a net of 10^5 neurons provided that the required nonlinear light amplifiers (photodetector/thresholding amplifier/LED or SLM driver array) become available. This is another instance where integrated optoelectronics technology can play a central role. We have built a 8 × 8 neuron version of the arrangement in Fig. 6(a) employing a square LED array, a square plastic lenslet array, and a square PDA, each of which has 8 × 8 elements in which the state update was computed serially by a computer which sampled the activation potentials provided by the PDA and furnished the drive signals to the LED array. The connectivity weights in this arrangement were stored in a photographic mask which was formed with the help of the system itself in the following manner: Starting from a set of unipolar binary matrices b_{ij} to be stored in the net, the required 4-D connectivity tensor was obtained by computing the sum of the outer products of the bipolar binary versions $v_{ij} = 2b_{ij} - 1$. The resulting connectivity tensor was partitioned and unipolar binary quantized versions of its submatrices were displayed in order by the computer on the LED display and stored at their appropriate locations in a photographic plate placed in the image plane of the lenslet array by blocking all elements of the lenslet array except the one where a particular submatrix was to be stored. This process was automated with the aid of a computer controlled positioner scanning a pinhole mask in front of the lenslet array so that the photographic plate is exposed to each submatrix of the connectivity tensor displayed sequentially by the computer. The photographic plate was then developed and positioned back in place. Although time-consuming, this method of loading the connectivity matrix in the net has the advantage of compensating for all distortions and aberrations of the system.

The procedure for loading the memory in the system can be speeded up considerably by using an array of minute electronically controlled optical shutters (switches) to replace the function of the mechanically scanned pinhole. The shutter array is placed just in front or behind the lenslet array such that each element of the lenslet array has a corresponding shutter element in register with it. An electronically addressed ferroelectric liquid crystal spatial light modulator (FeLCSLM) (see Spatial Light Modulators and Applications in list of further reading) is a suitable candidate for this task because of its fast switching speed (a few microseconds). Development of FeLCSLMs is being pursued worldwide because of their speed, high contrast, and bistability which enables nonvolatile switching of pixel transmission between two states. These features make FeLCSLMs also attractive for use as programmable connectivity masks in learning networks such as the Boltzmann machine in place of the MOSLM presently in use.

Because the connectivity matrix was unipolar, an adaptive threshold equal to the mean or energy of the iterated state vector was found to be required in computing the update state to make the network function as an associative

memory that performed in accordance with theoretical predictions of storage capacity and for successful associative search when sketchy (noisy and/or partial) inputs are presented. Recent evidence in our work is showing that ligistic neurons, mentioned in a footnote earlier, allow using unipolar connectivity weights in a network without having to resort to adaptive thresholding. This behavior may be caused by the possibility that logistic neurons, with their "humped" nonsigmoidal response, combine at once features of excitatory and inhibitory neurons which, from all presently available evidence, is biologically not plausible. Biological plausibility, it can be argued, is desirable for guiding hardware implementations of neural nets but is not absolutely necessary as long as departures from it facilitate and simplify implementations without sacrificing function and flexibility.

Several variations of the above basic 2-D architecture were studied. One, shown in Fig. 6(b) employs an array of light integrating elements (lenslet array plus diffusers, for example) and a CCD camera plus serial nonlinear amplification and driving to display the updated state matrix on a display monitor. In Fig. 6(c) a microchannel spatial light modulator (MCSLM) is employed as an electron-optical array of thresholding amplifiers and to simultaneously display the updated state vector in coherent laser light as input to the system. The spatial coherence of the state vector display in this case also enables replacing the lenslet array with a fine 2-D grating to spatially multiplex the displayed image onto the connectivity photomask. Our studies show that the 2-D architectures described are well suited for implementing large networks with semi-global or local rather than global interconnects between neurons, with each neuron capable of communicating with up to few thousand neurons in its vicinity depending on lenslet resolution and geometry. Adaptive learning in these architectures is also possible provided a suitable erasable storage medium is found to replace the photographic mask. For example in yet another conceivable variant of the above architectures, the lenslet array can be used to spatially demultiplex the connectivity submatrices presented in a suitable Z-D erasable display, i.e. project them in perfect register, onto a single SLM device containing the state vector data. This enables forming the activation potential array u_n directly and facilitates carrying out the required neron response operations (nonlinear gain) optically and in parallel through appropriate choice of the state vector SLM and the architecture. Variations employing internal feedback, as in 1-D neural nets, can also be conceived.

Discussion

Optoelectronics (or photonics) offers clear advantages for the design and construction of a new generation of analog computers (neurocomputers) capable of performing computational tasks collectively and dynamically at very high speed and as such, are suited for use in the solution of complex problems encountered in cognition, optimization, and control that have defied efficient handling with traditional digital computation even when very powerful digital computers are used. The architectures and proof of concept prototypes described are aimed at demonstrating that the optoelectronic approach can combine the best attributes of optics and electronics together with programmable nonvolatile spatial light modulators and displays to form versatile neural nets with important capabilities that include associative storage and recall, self organization and adaptive learning (self-programming), and fast solution of optimization problems. Large-scale versions of these neurocomputers are needed for tackling real world problems. Ultimately these can be realized using integrated optoelectronic (integrated photonic) technology rather than the hybrid optoelectronic approach presented here. Thus, new impetus is added for the development of integrated optoelectronics besides that coming from the needs of high speed optical communication. One can expect variations of integrated optoelectronic repeater chips utilizing GaAs on silicon technology being developed with optical communication in mind (see J. Shibata and T. Kajiwara in list of further reading). These, when fabricated in dense array form, will find widespread use in the construction of large-scale analog neurocomputers. This class of neurocomputers will probably also find use in the study and fast simulation of nonlinear dynamical systems and chaos and its role in a variety of systems.

Biological neural nets were evolved in nature for one ultimate purpose: that of maintaining and enhancing survivability of the organism they reside in. Embedding artificial neural nets in man-made systems, and in particular autonomous systems, can serve to enhance their survivability and therefore reliability. Survivability is also a central issue in a variety of systems with complex behavior encountered in biology, economics, social studies, and military science. One can therefore expect neuromorphic processing and neurocomputers to play an important role in the modeling and study of such complex systems especially if integrated optoelectronic techniques can be made to extend the flexibility and speed demonstrated in the prototype nets described to large scale networks. One should also expect that software development for emulating neural functions on serial and parallel digital machines will not continue to be confined, as at present, to the realm of straightforward simulation, but spurred by the mounting interest in neural processing, will move into the algorithmic domain where fast efficient algoriths are likely to be developed, especially for parallel machines, becoming to neural processing what the FFT (fast Fourier transform) was to the discrete Fourier transform. Thus we expect that advances in neuromorphic analog and digital signal processing will proceed in parallel and that applications would draw on both equally.

Acknowledgement

This overview derives from work conducted over the past five years under ARO, DARPA, JPL, ONR, and NSF sponsorship.

List of Further Reading

J. Alspector and R. B. Allen, "A Neuromorphic VLSI Learning System," in *Advanced Research in VLSI*, Paul Losleben, Ed., (MIT Press, Cambridge, MA, 1987).

N. Farhat, "Optoelectronic Analogs of Self-Programming Neural Nets..." *Applied Optics*, 26, 5093, 1987.

N. Farhat and D. Psaltis, "Optical Impelmentation of Associative Memory" in *Optical Signal Processing*, J. L. Horner, Ed., Academic Press, 1987, pp. 129–162.

N. Farhat, D Psaltis, A. Prata and E. G. Paek, "Optical Implementation of the Hopfield Model," *Applied Optics*, 24, 1469, 1985.

N. Farhat and Z. Y. Shae, "An Optical Learning Machine," *Proc. 1989 Hawaii International Conference on System Science*, Vol. I, IEEE Computer Society Press, IEEE Cat. No. 89TH0242-8,432 (1989).

S. Grossberg, *Studies of Mind and Brain*, Reidel, Boston, 1982.

D. O. Hebb, *The Organization of Behavior*, J. Wiley, New York 1949.

G. E. Hinton, and T. J. Sejnowski, "Learning and Relearning in Boltzmann Machines," in *Parallel Distributed Processing*, D. E. Rumelhart and J. L. McClelland (Eds.), Vol. I, Bradford-MIT Press, Cambridge, MA, 1986.

J. J. Hopfield, "Neural Networks and Physical Systems with Emergent Collective Computational Abilities," *Proc. Natl. Acad. Sci.* 79, 2554, 1982; "Neurons with Graded Response Have Collective Computational Properties Like Those of Two-State Neurons," *Proc. Natl. Acad. Sci.* 81, 3088, 1984.

R. J. McEleice, E. C. Posner, E. R. Rodemich and S. S. Venkatesh, "The Capacity of the Hopfield Associative memory," *IEEE Transactions on Information Theory*, Vol. IT-33, 461–482, 1987.

K. Nakano, "Associatron – a Model of Associative Memory," *IEEE Trans. Syst. Man Cybern.* SMC-2, 380, 1972.

D. B. Parker, "Learning Logic," MIT Tech. Report TR-47, 1985.

D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning Internal Representation by Error Propagation," in *Parallel Distributed Processing*, D. E. Rumelhart and J. L. McClelland (Eds.), Vol. I, Bradford-MIT Press, Cambridge, MA, 1986.

J. Shibata and T. Kajiwara, *IEEE Spectrum*, 26, 34, 1989.

"Spatial Light Modulators and Applications," *Optical Society of America Technical Digest Series*, Vol. 8, 1988.

P. Werbos, "Beyond Regression: New Tools for Prediction and Analysis in the Behavioral Sciences," Harvard University Dissertation, 1974.

R.J. Marks II is the current chairman pro term of the *IEEE Neural Networks Committee*. He was co-founder and first chairman of the *IEEE Circuits and Systems Technical Committee on Neural Systems and Applications*. He is a Professor of Electrical Engineering at the University of Washington, Seattle. He was asked by the LEOS editor of *Circuits and Devices Magazine* to be the guest editor of this series of three articles on neural networks.

Nabil H. Farhat received the B.Sc. degree in 1957 from the Technion-Israel Institute of Technology, Haifa, the M.Sc. degree in 1959 from the University of Tennessee, Knoxville, and the Ph.D. degree in 1963 from the University of Pennsylvania, Philadelphia – all in Electrical Engineering.

In 1964 he joined the faculty of the Moore School of Electrical Engineering, University of Pennsylvania, where he is now Professor in Electrical Engineering and heads the Electro-Optics and Microwave-Optics Laboratory. His research encompasses: image understanding, microwave imaging and holography, optical information processing and modeling of neural network and self-organizing systems for neurodynamical pattern and target identification in all of which he has published extensively. His teaching covers EM theory, electron and light optics, neurodynamics, optics, and optical computing.

While being an Associate Professor, Dr. Farhat held the Ennis Chair in Electrical Engineering. He has held visiting professorships at Universities in the U.S. and abroad. In 1985 he was named Distinguished Visiting Scientist at the Jet Propulsion Laboratory in Pasadena and has served in that capacity since then for several summer periods. He is a recipient of the University of Pennsylvania Christian R. and Mary F. Lindback Foundation award for distinguished teaching, is a Fellow of the Institute of Electrical and Electronics Engineers, and the Optical Society of America, and member of several other professional and honorary societies. He also served on the National Board of Directors of Eta Kappa Nu and was an RCA consultant from 1969 to 1980. Dr. Farhat was Editor of *Advances in Holography* and Associate Editor of *Acoustical Imaging and Holography*. He is presently action editor for two neural network journals: *Neural Networks* and *Neural Computation*.

Appendix II

AN OPTICAL LEARNING MACHINE

Nabil H. Farhat
Zon-Yin Shae

Reprinted from PROCEEDINGS OF THE TWENTY-SECOND ANNUAL HAWAII INTERNATIONAL CONFERENCE ON SYSTEM SCIENCE
Kailua-Kona, Hawaii, January 3-6, 1989

COMPUTER SOCIETY PRESS REPRINT

The Computer Society of the IEEE
1730 Massachusetts Avenue NW
Washington, DC 20036-1903

Washington • Los Alamitos • Brussels


IEEE

THE INSTITUTE OF ELECTRICAL AND ELECTRONICS ENGINEERS, INC.

COMPUTER SOCIETY PRESS

AN OPTICAL LEARNING MACHINE

Nabil H. Farhat and Zon-Yin Shae

University of Pennsylvania
Electrical Engineering Department
Philadelphia, PA 19104-6390

Abstract

We report on what we believe to be the first demonstration of a fully operational optical learning machine. Learning in this machine is stochastic taking place in a self-organizing tri-layered opto-electronic neural net with plastic connectivity weights that are formed in a programmable non-volatile spatial light modulator. The net learns by adapting its connectivity weights in accordance to environmental inputs. Learning is driven by error signals derived from state-vector correlation matrices accumulated at the end of fast annealing bursts that are induced by controlled optical injection of noise into the network. Operation of the machine is made possible by two developments in our work: Fast annealing by optically induced tremors in the energy landscape of the net, and stochastic learning with binary weights. Details of these developments together with the principle, architecture, structure, and performance evaluation of the machine are given. These show that a 24 neuron prototype machine can learn, with a learning score of about 53%, to associate three 8-bit vector pairs in 10-60 minutes with relatively slow (60 msec response time) neurons and that shifting to neurons with 1 μsec response time for example, would reduce the learning time by roughly 10^4 times. Methods for improving the learning score presently under study are also discussed.

1. Introduction

Ever since the fit between what neural net models can offer (collective, iterative, nonlinear, robust, and fault-tolerant approach to information processing) and the inherent capabilities of optics (parallelism and massive interconnectivity) was first pointed out [1],[2] and the first optical associative memory demonstrated in 1985 [3], work and interest in neural net analogs and neuromorphic optical signal processing has been growing steadily (see for example [4]-[10]). In addition to the vector-matrix multiplication with thresholding and feedback scheme utilized in early implementations, an arsenal of sophisticated optical tools such holographic storage, phase conjugate optics, and wavefront modulation and mixing are being drawn upon to realize associative memory functions. Such functions include auto-associative, hetero-associative, storage and recall [1]-[9], with signal recovery and pattern recognition from partial information receiving much attention as potential application [6],[11].

It is becoming increasingly clear, however, that associative memory is only one apparent function of biological neural nets that lends itself to optical implementation. Optics can play a useful role in the implementation of artificial neural nets capable of self-organization and learning i.e., self-programming nets [12]-[14]. One can safely state that self-organization and learning is the most distinctive single feature that sets neuromorphic processing apart from other approaches to information processing. Learning in these nets is by adaptive modification of the weights of interconnections between neurons (plasticity). It can be supervised or unsupervised, deterministic or stochastic.

Self-organization and learning in multilayered neural nets is being studied because of the promise of developing machines that can program themselves with nominal supervision alleviating thereby the programming complexity associated with massively parallel and distributed computation systems. Here we are concerned with fine-grained parallelism where computation is performed through collective dynamical behavior of a large number of simple interconnected switching elements (neurons). Self-organization in such nets can be deterministic as in the error back-propagation algorithm [15] or stochastic as by simulated annealing [16],[17] within the framework of a Boltzmann machine [18], [19] and by a probabilistic extension of the error back-projection algorithm [20].

In this paper we concern ourselves with stochastic learning because: (a) learning machines that learn from environmental representations are expected to operate in random or fuzzy environment that can best be described probabilistically. Such machines learn the probability distribution function of their environmental inputs, (b) to gain an understanding of the role of noise in complex dynamical systems such as the nervous system, (c) since learning in these machines involves finding the global minimum of a cost or penalty function (error driven learning) they can be also used to solve combinatorial optimization problems whose solution requires also finding the state of global minimum of a cost function.

In the following, the principle, architecture, and methodology of stochastic learning in an opto-electronic setting are presented in Section 2. This is followed by discussion in Sections 3 and 4 respectively of the noisy thresholding and stochastic

0073-1129/89/0000/0432$01.00 © 1989 IEEE

learning with binary weights schemes utilized. The results of numerical simulation and experimental verification of the two schemes in opto-electronic hardware are given in Sections 5 and 6 and are followed by a brief discussion of these results and their implications.

2. Stochastic Learning and Machine Architecture

Optics and opto-electronic architectures and techniques can play an important role in the study and implementation of self-programming networks and in speeding-up the execution of relevant learning algorithms. Learning requires partitioning a net into layers with a prescribed communication pattern among them. A method for partitioning an opto-electronic analog of a neural net into input, output, and internal groups (layers) of neurons with prescribed communication pattern among neurons within each layer and between layers that is capable of stochastic learning, by means of a simulated annealing algorithm in the context of a Boltzmann machine formalism, has been described earlier [18]. A schematic of the opto-electronic network involved is given in Fig. 1(a). The network, consisting of

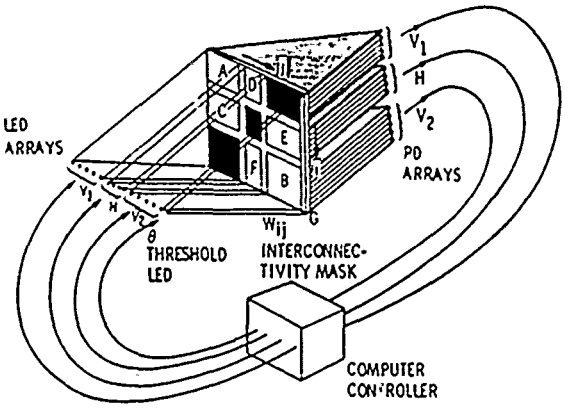


Fig. 1. Architecture for opto-electronic analog of layered self-programming net.

N neurons, is partitioned into three groups. Two groups, V_1 and V_2, represent visible or environmental units that can be used as input and output units respectively. The third group H are hidden units. The partition is such that $N_1 + N_2 + N_3 = N$ where subscripts 1,2, and 3 on N refer to the number of neurons in the V_1, V_2 and H groups respectively. The interconnectivity matrix, designated here as w_{ij}, is partitioned into nine submatrices, A, B, C, D, E, and F plus three zero submatrices shown as blackened or opaque regions of the w_{ij} mask. The LED array represents the state of the neurons, assumed to be unipolar binary (LED on = neurons firing, LED off = neurons not-firing). The w_{ij} mask represents the strengths interconnections between neurons. Light from the LEDs is smeared vertically over the w_{ij} mask with the aid of an anamorphic lens system (not shown in Fig. 1(1)) and light emerging from rows of the mask is focused with the aid of another anamorphic lens system (also not shown) onto elements of the photodetector (PD) array. Bipolar values of w_{ij} can be realized in incoherent light by separating each row of the w_{ij} mask into two subrows and assigning positive values of w_{ij}^{+} to one subrow and negative values w_{ij}^{-} to the other, then focusing light emerging from the two subrows separately onto pairs of adjacent photosites connected in opposition in each of the V_1, V_2 and H segments of the PD array as described elsewhere [2]. Submatrix A, with $N_1 \times N_1$ elements, provides the interconnection weights of units or neurons within group V_1. Submatrix B, with $N_2 \times N_2$ elements, provides the interconnection weights of units within V_2. Submatrices C (of $N_1 \times N_3$ elements) and D (of $N_3 \times N_1$ elements) provide the interconnection weights between units of V_1 and H and similarly submatrices E (of $N_2 \times N_3$ elements) and F (of $N_3 \times N_2$ elements) provide the interconnection weights of units V_2 and H. Units in V_1 and V_2 can not communicate with each other directly because locations of their interconnectivity weogjts om tje w_{ij} matrix or mask are blocked out (blackened lower left and top right portion of w_{ij}). Similarly units within H do not communicate with each other because locations of their interconnectivity weights in the w_{ij} mask are also blocked out (center blackened square of w_{ij}). The LED element θ is of graded response. It can be viewed as representing the state of an auxiliary neuron in the net that is always on to provide a threshold level to all units by contributing to the light focused onto only negative photosites of the PD array by suitable modulation of pixels in the *G* column of the interconnectivity mask. This method for introducing the threshold level is attractive as it allows for introducing a fixed threshold to all neurons or an adaptive threshold if desired. It can also be employed to alter the energy landscape of the net adaptively in accordance to the behavior of other parameters of the net. A computer works as the system controller to calculate P_{ij} and P'_{ij}, and also to control the MOLSM which implements the interconnectivity matrix *W*. This architecture allows stochastic learning by simulated annealing in the context of a Boltzmann machine. The learning algorithm for Boltzmann machine can be summarized as follows:

1. Choose one mapping or associated pair that the net is required to learn, and present it to the net. The associated pair consists of two unipolar binary vectors one an input vector and the other an output vector.

2. Clamp the input vector to the V_1 neurons, and the corresponding output vector to the V_2 neurons.

3. Employ simulated annealing method in energy space as described in [18] to find low energy configurations for the given V_1 and V_2. The final temperature in the cooling schedule is called T_c and will be used later as an annealing parameter in Cross-Entropy or *G*-space. During this step, random drawing and change of only the states of the hidden neurons (H) takes place.

4. Repeat steps (2-3) *N1* times for all associations the net is required to learn, and collect co-occurrence statistics i.e. determine the

probabilities P_{ij} of the jth being in the same state i.e. both being on or off.

5. Unclamp the V_2 neurons and repeat steps 3-4 for all input vectors, and collect co-occurrence statistics again i.e. determine the probabilities P'_{ij} of the ith and jth neurons being in the same state. During this step, random drawing and change of both the states of the H and the V_2 neurons takes place.

6. All weights in the net are modified by increasing the synaptic weight (W_{ij}) between the ith and jth neurons by a small amount of δ if $(P_{ij} - P'_{ij} > 0$, and otherwise, decreasing the weight by the same amount. Note this requires multivalued W_{ij} or incremental variation of W_{ij} that requires the use of graded response spatial light modulators for realizing synaptic modifications in opto-electronic implementations.

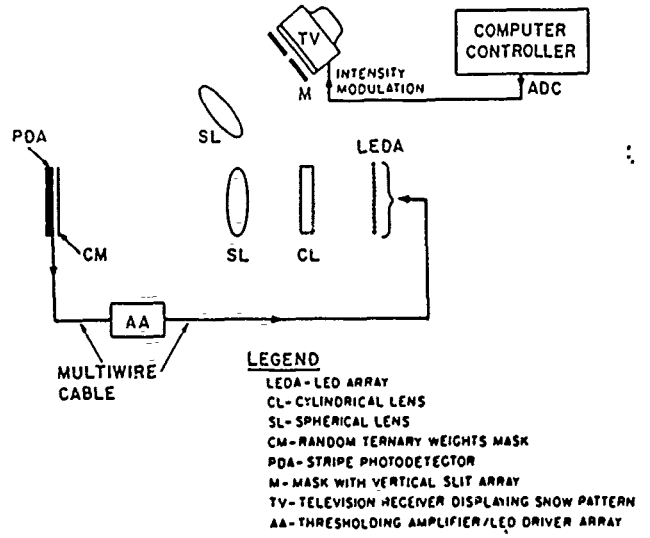
7. We call steps 1-6 a learning cycle. The learning cycle consists of two phases. Phase one involves clamping the input and output units to the associated pairs. Phase two involves clamping the input vector alone and letting the output units free run with the hidden units. The learning cycle is repeated again and again and is halted after $(P_{ij} - P'_{ij})$ is close to zero for every i and j.

The learning procedure described above can be supported in the opto-electronic hardware environment described previously. However, the above procedures can be considerably simplified and accelerated by exploiting the inherent capabilities of the opto-electronic approach as is described in the next sections.

3. Fast Annealing by Noisy Thresholding

A spatially and temporally uncorrelated linear array of perculating light spots of suitable size and intensity range can be generated and imaged onto the PD array of Fig. 1 directly such that both the positive and negative photosites of the PD array are subjected to random irradiance. This introduces a random (noise) component in the threshold levels of the neurons. The noisy threshold produces in turn a noisy component in the energy function of the net. The magnitude of the noise components can be controlled by varying intensity of the light spots array irradiating the PD array. The noisy threshold produces therefore random controlled perturbation or "shaking" of the energy landscape of the net. This helps shake the net loose whenever it tends to get trapped in a local energy minimum. The procedure can be viewed as that of generating controlled gradually decreasing deformations or tremors in the energy landscape of the net that prevents entrapment in a local energy minimum and helps the net settle into the global minimum energy state or one close to it and stay there. Both the random drawing of neurons (more than one at a time is now possible) and the stochastic state update of the net are now done in parallel at the same time and without having to compute the change in the energy of the net and associated Boltzmann factor as required ordinarily in simulated annealing algorithms. This leads to significant acceleration of the annealing process. Electronic control of the random light array intensity enables realizing any annealing profile. We have presented the results of numerical study of this noisy thresholding scheme elsewhere [21] demonstrating that it can perform equally well as conventional simulated annealing with some advantage in as far as the number of iterations needed to find a global energy minimum is lower. In the following, results of an experimental study and verification of the scheme are presented.

An annealing experiment (see Fig. 2) based on the noisy threshold algorithm in an opto-electronic neural net was devised. The "snow" pattern displayed by a television receiver tuned to an empty



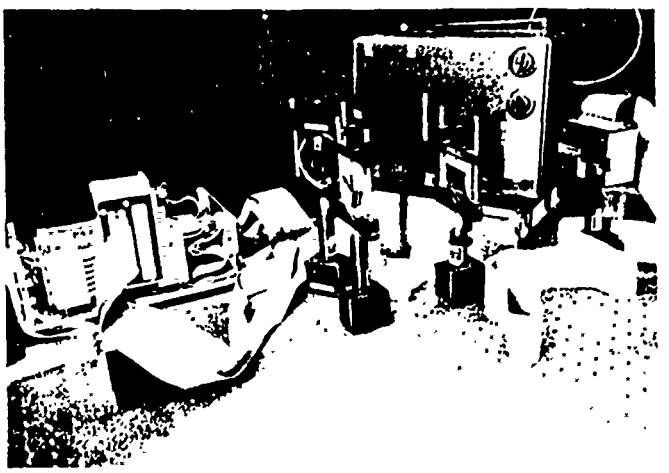


Fig. 2. Schematic representation and pictorial view of opto-electronic scheme used to verify fast annealing by noisy thresholding in a stochastic neural net.

channel is used as the spatio-temporal optical noise source. We use a lens to project a portion of the snow pattern onto the photodetector array PDA of an opto-electronic neural net consisting of 16 unipolar binary neurons of the type described elsewhere [3]. The connectivity matrix of the network was the same random ternary matrix utilized in earlier work [21]. The brightness of the TV screen is controlled by the D/A output of a MASSCOMP computer, and the convergent state is monitored by the A/D input of the same computer. We investigated four types of cooling profiles: linear, concave, convex, and stair-case illustrated in Figure 3. For each cooling profile, we tested 5 annealing time invervals: 100, 200, 500, 1000, and 2000 ms. For each cooling profile and annealing time interval, we do the annealing 100 times to collect sufficient statistics, and find the probability that the system converges to the state of global energy minimum or close to it. The experimental results obtained show that the system can find the global energy minimum of an artificial neural net of 16 neurons in 2000 ms which correspond to 32 time constants of the neurons in the test network. A net of neurons with response time of 1 μ sec would anneal therefore in few tens of microseconds and this is expected to be independent of the number of neurons in the net as long as parallel injection of noise in the network is implemented. The cooling profile has no observable effect on this result. The probabilities of convergence to a global minimum as function of the annealing duration for different annealing profiles are shown in Table 1.

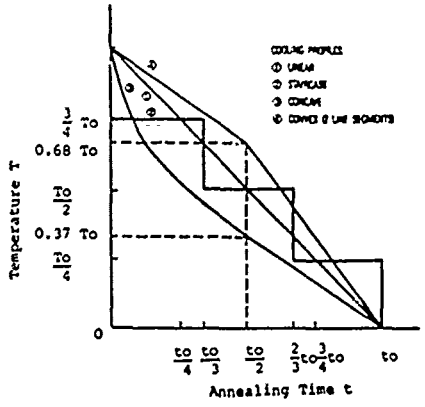


Fig. 3. Cooling profiles

Table 1. The probabilities of convergence to a global minimum as function of the annealing duration for different annealing profiles.

	linear	concave	convex	step
100ms	0.48	0.46	0.50	0.45
200ms	0.62	0.56	0.68	0.64
500ms	0.78	0.73	0.77	0.79
1000ms	0.83	0.86	0.84	0.88
2000ms	0.97	0.96	0.96	0.98

4. Stochastic Learning with Binary Weights

The Boltzmann machine's learning algorithm described earlier employs graded weights. However, from practical viewpoint, learning in artificial neural nets can be simplified considerably if binary weights can be used. This would pave the way to using fast nonvolatile binary spatial light modulators (SLMs) such as Magneto-Optic SLMs and Ferroelectric liquid crystal SLMs. However, a Boltzmann machine basically is an adaptive system. If the step size of adaptive changes is too large and the sensitivity of system response to the error signal is high, the machine will generally become unstable. Since a traditional Boltzmann machine has ordinarily high sensitivity to the error signal, i.e., it responds to the error signal ($P_{ij} - P'_{ij}$) by modifying synaptic weights even when the error signal is very small, small weight variations are essential to prevent the system from becoming unstable. However, in a binary weight net (W_{ij} = 1, -1) the step size of adaptive change is large and fixed (-2 or 2). In order to prevent the system from becoming unstable, we increase the inertia of weights i.e. weights do not change when small value of $P_{ij} - P'_{ij}$ occurs. As a result, the learning procedure of the Boltzmann machine in a binary weight net would be identical to the procedure of the graded weights net stated in the system architecture section, except step 6 which is modified as follows: If $(P_{ij} - P'_{ij}) \geq M$, set W_{ij} = 1; if $(P_{ij} - P'_{ij}) \leq -M$, set W_{ij} = -1; otherwise no changes, where $M \in [0,1]$ is a fixed constant.

The goal of the Boltzmann machine is to minimize the Cross-Entropy G by modifying the weights of the net in a certain order. The G functional is an information theoretic measure of the distance between the probability distributions when an environmental input is present in the net and when it is free running with no or partial environmental input applies, and is given by

$$G = \sum_{\alpha} P^{+}(V\alpha) \ln \frac{P^{+}(V\alpha)}{P^{-}(V\alpha)} \qquad (1)$$

where $P^{+}(V\alpha)$ is the probability of the visible units being in the α state when the visible units are subjected to the environmental input. Namely, $P^{+}(V\alpha)$ represents the desired or specified probability for the α state. $P^{-}(V\alpha)$ is the corresponding probability when the net is free-running. Namely, $P^{-}(V\alpha)$ represents the actual probability generated from the net for the α state. $P^{-}(V\alpha)$ depends on the weights W_{ij}, and so G can be altered by changing W_{ij}. Since, in general, there are local minima in G space, a gradient descent search will find a local minimum instead of the global minimum. In order to reach the global minimum in G space introduction of noise in G space is required. However, if the noise level is too large, the network can not learn the specified

or desired environmental distribution. A systematic way for adding noise in G space, i.e. an annealing scheme in G space, has not yet been studied in detail. Here we propose the use of the final temperature T_O of the simulated annealing schedule used in the energy space E as the annealing parameter in G space, since $P^-(V\alpha)$ is function of W_{ij} and hence on T_O. In the first few learning cycles, we use high values of T_O. This will provide high level of noise in G space. The value of T_O is decreased gradually along with the number of learning cycles. Accordingly, a simulated annealing process in G space is realized by decreasing the final temperature T_O in a similar way to the simulated annealing process in energy space which is accomplished by decreasing the annealing temperature T. Note also that an annealing schedule G-space with high values of T_O is equivalent to a short time interval annealing schedule in E space, i.e., both cases can generate high level of noise in G space, and vice versa. Accordingly, the annealing time interval in E space can also be used as an annealing parameter in G space. As a result, a simulated annealing process in G space can also be accomplished by gradually increasing the annealing time interval in E space along with the number of learning cycles. Results of computer simulations of stochastic learning by simulated annealing in a Boltzmann machine employing both graded and binary weights are presented in the next section.

5. Simulation Results

In these simulations we use the noisy threshold (N-T) annealing scheme and use the annealing time interval in E space as an annealing parameter in G space. All the simulations learn to solve a 4-2-4 encoder problem [18] in the context of Boltzmann machine formalism i.e. this consists of having a three layered net, of the kind described in the architecture section, learn to form its own internal representations of the associations presented to it. For all simulations, the net reaches equilibrium 100 times (25 times for each input vector) for collecting the statistics of P_{ij} during the input and output clamping phase. The situation is the same for collecting the statistics of P'_{ij}. All annealing schedules are stated in the corresponding Figures in the notation of $I@T$ designating the number of iterations I at each temperature value T. The noise we used is binary noise whose amplitude is either T or $-T$ and is decreased gradually in time and terminated at T_O. Figure 4 shows the results of the linear weight learning scheme, and Fig. 5 shows the results of the binary weight learning scheme when the parameter M we used was 0.1. Both figures show the results of 12 runs. Only two annealing schedules in E space of different time constants are used for the annealing in G space. During the first half of the total number of learning cycles the short time interval annealing schedule is employed, and during the later half of the learning cycles the long time interval annealing schedule is employed. These results demonstrate that annealing in G space is possible, and also show that stochastic learning with binary weights

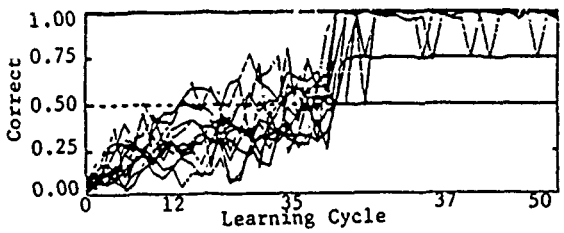


Fig. 4. Linear weight learning curve with N-T algorithm. Annealing schedule in E space: during the 0-25th learning cycle 2 @ 3, 1 @ 1.5, 1 @ 1, and 2 @ 0.1; during the 26-50th learning cycle 6 @ 1, 6 @ 0.8, 6 @ 0.5, 6 @ 0.1, and 6 @ 0. This is a annealing scheme in G space.

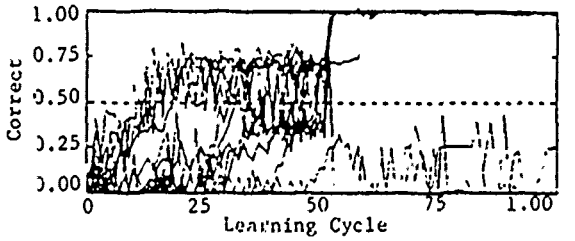


Fig. 5. Binary weight learning curve with N-T algorithm. Annealing schedule in E space: during the 0-50th learning cycle 2 @ 3, 1 @ 1.5, 1 @ 1, and 2 @ 0.1; during the 51-100th learning cycle 4 @ 1, 4 @ 0.8, 4 @ 0.5, 4 @ 0.1, and 4 @ 0. This is a annealing scheme in G space.

is possible provided that inertia is introduced in the weights update rule. It is worth noting that not all learning trials in Figs. 4 and 5 fully succeed and this, as will discussed later, determines the effectivenes. of learning in the network.

6. Hardware Implementation

A top view of the layout of an optoelectronic stochastic optical learning machine consisting of 24 unipolar binary neurons is shown in Fig. 6. The net is partitioned into three layers as in the architecture described earlier (see Fig. 1). It utilizes a computer controlled 48 x 48 pixel magneto-optic SLM (MOSLM) to realize bipolar binary synaptic weight modification. The state vector of the network is displayed by a 24 LED array (LEDA). Elements in this array belonging to groups V_1 and V_2 can be clamped into fixed prescribed states for any desired duration and unclamped from them by the computer controller during learning. The usual type of anamorphic lens systems (CL, SL) are used to project the state vector onto pixels of the MOSLM and also to focus light emerging from the 48 rows of the MOSLM onto 48 elements of the photodetector array (PDA). Pairs of the PDA elements, connected in opposition, measure the activation potentials of the neurons in the net in a manner similar to that described in [3]. Because of the

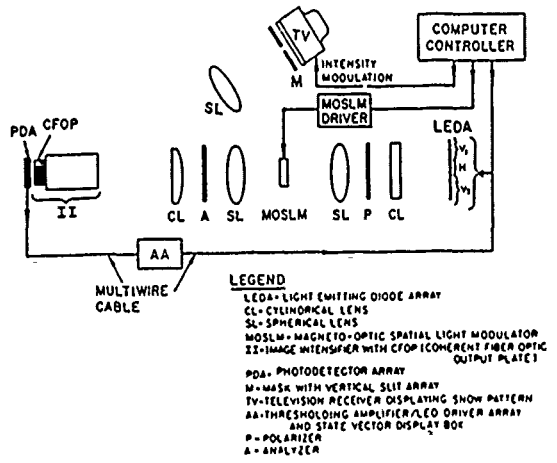


Fig. 6. Opto-electronic stochastic learning machine.

relatively high transmission loss of the MOSLM/ crossed polarizers (P, A) combination, an image intensifier (II) is employed to amplify the light pattern emerging from the MOSLM as seen through the output anamorphic lens system. Examples of intensified versions of patterns stored in the MOSLM and projected directly onto the image intensifier (i.e. with the cylindrical lens LC in the output anamorphic lens system removed) are shown in the top row of Fig. 7. In the bottom row of Fig. 7, are shown horizontally compressed versions of these patterns obtained when the cylindrical lens LC was reinserted. These compressed patterns are proximity coupled to the stripe elements of the PDA to form the activation potentials of the neurons as described earlier. A bank of 24 differential thresholding amplifiers/LED drivers are used to form from the activation potentials the state vector of the net which is then displayed by the LEDA and acts as input to the net to complete the feedback iteration and interconnection between neurons.

7. Results

The results reported in this section were obtained by using a variation of the G-space annealing scheme employed in the simulations of the 4-2-4 encoder problem discussed in Section 5. As in the scheme described there, we again generate noise in G-space by using a short annealing time interval in E-space, but instead of increasing the E-space annealing time interval with learning cycle in order to gradually reduce the G-space noise and achieve G-space annealing, we keep the short annealing time interval fixed. We have found that although the annealing schedule in E-space is now the same for all learning cycles, the amount of effective noise in G-space decreases gradually and automatically because of the process of self-organization taking place in the net as learning proceeds. Learning begins when the initial connectivity matrix is random. Thus, during the first learning cycles the interconnectivity matrix is random. The basins of attraction in phase-space of the net for the vectors being learned have not had a chance to develop yet. The activation potentials of neurons during this phase are close to zero. Therefore the optically injected noise causes a relatively high effective noise level in the net and a short duration annealing schedule is not able mostly to find the global energy minimum of the net. However, as the net gradually self-organizes and basins of attraction for the vectors being stored develop, the effectiveness of the short annealing schedule in finding the global energy minimum improves. This corresponds to a decrease in noise level in G-space and to an effective annealing schedule in G-space. The results of software and hardware simulation presented below show the effectiveness

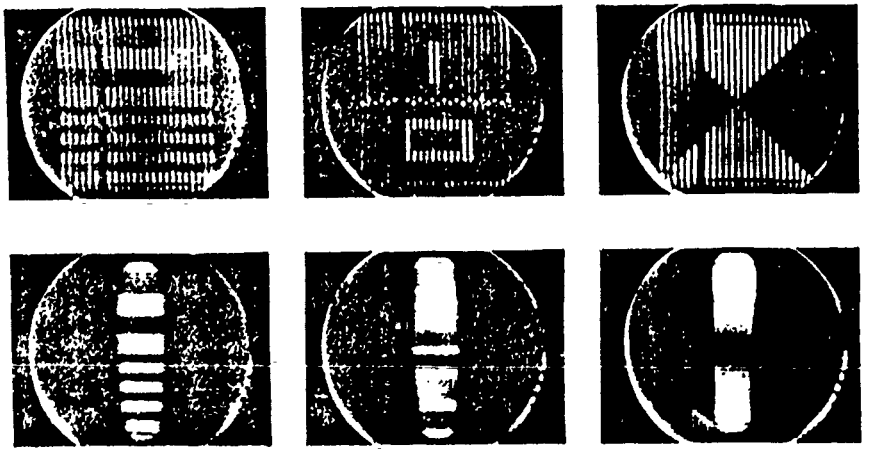


Fig. 7. Intensified patterns stored in the MOSLM (top row) and horizontally compressed versions (bottom row)

of the fixed schedule annealing in G-space scheme.

In the software simulations, the net is trained to learn three self-mappings or auto-associations involving three 8 bit vector pairs:

V_1	V_2
10100000	10100000
00010010	00010010
00000101	00000101

with the following E-space annealing schedule: 3@3, 3@2, 4@1, 10@0. The injected noise level was uniformly distributed in (-T, T), T being the annealing temperature. We run the simulation 40 times, and each run consists of 100 learning cycles. For all simulations, the net reaches equilibrium 9 times (3 times for each input vector) for collecting the statistics of P_{ij} during the input and output clamping phase. The same number and pattern of runs is followed in collecting the statistics of P'_{ij}. The value of M used is equal to 2/9. A learning run is considered successful if the net can learn the desired 3 mappings in the 100 learning cycles. There are 25 runs of successful learning out of the 40 runs of simulations corresponding to a learning score is therefore 62.5%. The learning curves are shown in Fig. 8 (top) with two individual typical learning curves shown in Fig. 8 (middle).

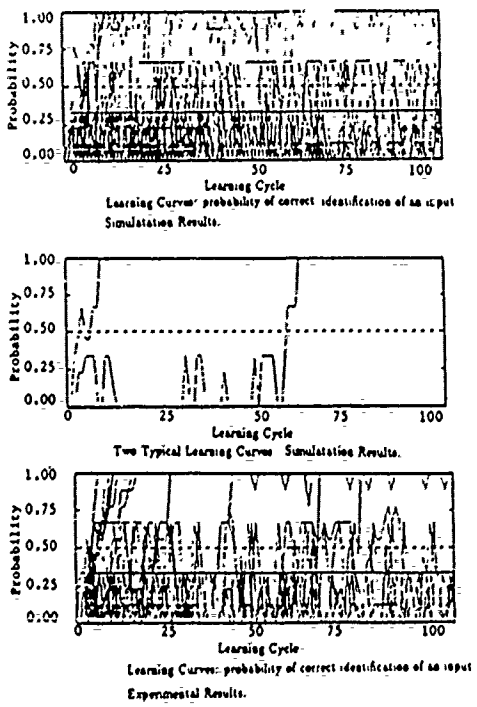


Fig. 8. Learning Performance

Next the hardware implementation described in Section 6 was employed to test learning the same three mappings or auto-associations used in the above simulation. E-space annealing is now realized however with a linear annealing schedule of 1 sec. duration. Because the net requires 2 sec. annealing time interval to reach a global minimum in E-space (see Table 1), a 1 sec. annealing time interval is found to be short enough to introduce noise in G-space. All other parameters are the same as in the numerical simulation case. We exercised the net 30 times and found 16 runs of successful learning out of 30 tries. This corresponds to a learning score of 53%. The learning curves obtained are shown in Fig. 8 (bottom) and typical individual learning curves looked very much like those of the simulation (Fig. 8 (middle)). The time required for the net or machine to learn the mappings with the above score ranged between 10 minutes to 60 minutes. This time is determined primarily by the annealing time interval utilized which depends on the time constant of the neurons in the network (60 msec for the prototype of Section 6). Assuming that faster neurons (e.g. 1 μsec neurons) and a suitable faster optical noise injection scheme are employed, the above learning time may be cut by a factor of about 10^4 and this is expected to be independent of the number of neurons in the network because of the inherent parallelism of the optically induced annealing scheme. Several schemes for improving the above learning score are presently under study. We find for example that reducing the number of associations to be learned from 3 to 2 and calculating the coincidence or co-occurrence probabilities P_{ij} and P'_{ij} by counting only on-on correlations in the state vectors of the net during learning and excluding off-off correlations, the learning score improves dramatically (to near perfect).

8. Conclusions

We have described an architecture for partitioning an opto-electronic analog of a neural net to form a multilayered net that permits self-organization and learning when computer controlled nonvolatile spatial light modulators are utilized to realize the required plasticity. The focus here is on stochastic learning as opposed to deterministic learning because: (a) this leads to machines that are inherently amenable to learning sketchy representations or feature spaces of practical environments that are best described probabilistically (probabilistic learning), and (b) this may provide useful insight in the role of noise in biological neural nets. We show that departure from the conventional simulated annealing algorithm through the use of noisy thresholding in opto-electronic schemes can markedly accelerate the annealing process, and make stochastic learning practical. Employing the noisy thresholding scheme, a small opto-electronic neural net (of 16 neurons) was found to reach a global energy minimum or one close to it in about 32 neuron time constants. We also show that binary weight learning algorithm can be used in the context of a modified Boltzmann machine. This paves the way to the use of nonvolatile binary spatial light modulators to realize the required plasticity in such stochastic learning nets. Such nets, having learned

their environmental inputs can be "frozen" for use as associative memories of the entities learned by merely removing injected noise from the net. Noise injection for annealing returns the nets to a "soft" mode for learning new environmental inputs.

9. Acknowledgement *

Research work reported here was supported by the Defense Advanced Projects Agency (Grant N000014-85-K-2036 P004) and the National Science Foundation (Grant EET 8716685). Partial support from the Jet Propulsion Laboratory (for later phases of this work) and the University of Pennsylvania's LRSM is also gratefully acknowledged.

10. References

[1] D. Psaltis and N. Farhat, "A New Approach to Optical Information Processing Based on the Hopfield Model," in Digest of the Thirteenth Congress of the International Commission on Optics (ICO-13), Sapporo, Japan (1984), p. 24.

[2] D. Psaltis and N. Farhat, "Optical Information Processing Based on an Associative-Memory Model of Neural Nets with Thresholding and Feedback," Opt. Lett. 10, 98 (1985).

[3] N.H. Farhat, D. Psaltis, A. Prata, and E. Paek, "Optical Implementation of the Hopfield Model," Appl. Opt. 24, 1469 (1985).

[4] J. Von Neuman, The Computer and the Brain, Yale University Press, New Haven, (1958).

[5] A.D. Fisher, "Implementation of Adaptive Associative Optical Computing Elements," Proc. Soc. Photo-Opt. Instrum. Eng. 625, 196 (1986).

[6] B.H. Soffer, G.J. Dunning, Y. Owechko, and E. Marom, "Associative Holographic Memory with Feedback Using Phase-Conjugate Mirrors," Opt. Lett. 11, 118 (1986).

[7] D.Z. Anderson, "Coherent Optical Eigenstate Memory," Opt. Lett. 11, 56 (1986).

[8] A. Yariv and S.K. Kwong, "Associative Memories Based on Message-Bearing Optical Modes in Phase-Conjugate Resonators," Opt. Lett. 11, 186 (1986).

[9] B. Kosko and C. Guest, "Optical Bidirectional Associative Memory," Proc. Soc. Photo-Opt. Instrum. Eng. 758, (1987), in press.

[10] M. Takeda and J.W. Goodman, "Neural Networks for Computation: Number Representations and Programming Complexity," App. Opt. 25, 3033, (1986).

[11] N. Farhat, S. Miyahara, and K.S. Lee, "Optical Implementation of 2-D Neural Nets and Their Application in Recognition of Radar Targets," in Neural Networks for Computing, J.S. Denker, Ed. (American Institute of Physics, New York, 1986), p. 146.

[12] K. Wagner and D. Psaltis, "Multilayer Optical Learning Networks," App. Optics, 26, 5061, (1987).

[13] N. Farhat, "Architectures for Optoelectronic Analogs of Self-Organizing Neural Nets," Opt. Lett., 12, 448, (1987).

[14] N. Farhat, "Optoelectronic Analogs of Self-Programming Neural Nets: Architectures and Methodologies for Implementing Stochastic Learning by Simulated Annealing," App. Opt. 26, 5043 (1987).

[15] D.E. Rumelhart, G.E. Hinton and R.J. Williams, "Learning Internal Representations by Error Propagation," in Parallel Distributed Processing, D.E. Rumelhart and J.L. McClelland, eds. (Bradford-MIT, Cambridge, Mass. 1986), Vol. 1.

[16] N. Netropolis, A. Rosenbluth, M. Rosenbluth, and A. Teller, "Equations of State Calculations by Fast Computing Machines," J. Chem. Phys., 21, 1087 (1953).

[17] S. Kirkpatric, C. Gelat Jr., and M. Vecchi, "Optimization by Simulated Annealing," Science, 220, 671, (1983).

[18] D.H. Ackley, G. Hinton, and T. Sejnowski, "A Learning Algorithm for Boltzmann Machines," Cognitive Sci., 1. 147, (1985).

[19] M. Derthick, "Variations on the Boltzmann Machine Learning Algorithm," Carnegie Mellon University Report, CMU-CS-84-120.

[20] E.A. Baum and F. Wilczek, "Supervised Learning of Probability Distribution by Neural Networks," (Private Comm.) (1988).

[21] N. Farhat and Z.Y. Shae, "Architecture and Methodologies for Self-Organization and Stochastic Learning in Opto-electronics Analysis of Neural Nets," Proc. IEEE First Annual Meeting Conference on Neural Networks, IEEE Cat. 87TH091-7, June 1987.

* This work was also supported by SDIO/ISTO. Inadvertently acknowledgement of this support was regrettably overlooked.

LEARNING NETWORKS FOR EXTRAPOLATION AND RADAR TARGET IDENTIFICATION

Baocheng Bai and Nabil H. Farhat
University of Pennsylvania
The Moore School of Electrical Engineering
Electro-Optics and Microwave-Optics laboratory
Philadelphia, PA 19104

ABSTRACT: The problem of extrapolation for near perfect reconstruction and target identification from partial frequency response data by neural networks is discussed. Because of ill-posedness, the problem has traditionally been treated with regularization methods. The relationship between regularization and the role of hidden neurons in layered neural networks is examined. As a result, we are able to set up a layered nonlinear adaptive neural network for performing extrapolations and reconstructions with excellent robustness. The results are then extended to neuromorphic target identification from a single "look" (single broadband radar echo). A novel approach for achieving 100% correct identification in a learning net with excellent robustness employing realistic experimental data is also given. The findings reported have potential for obviating the need for forming radar images in order to identify targets and could furnish a viable and economical means for identifying non-cooperative targets.

1 Introduction

For an object function $o(r)$ of finite spatial extent, the corresponding frequency response $F(p)$ extends over the entire frequency domain $-\infty < p < +\infty$. Because of practical constraints, the frequency response $F(p)$ can only be measured in practice over a finite frequency window $p_1 < p < p_2$ to give the measured frequency response $F_m(p)$. The traditional and widely used approach of Fourier inversion, by means of a discrete Fourier Transform (DFT), as an algorithm for retrieving $o(r)$ from $F_m(p)$, violates *a priori* knowledge of the

object function and yields an estimate of $o(r)$ with limited resolution which may not satisfy resolution requirement in demanding applications.

More sophisticated methods for retrieving a better estimate of $o(r)$ from $F_m(p)$ exist. The retrieval of $o(r)$ from the partial information $F_m(p)$ in the presence of noise is known however to be an ill-posed problem[1],[2]. Studies[3] have been carried out for retrieving $o(r)$ by incorporating *a priori* knowledge and minimizing a certain "cost function" related to $F_m(p)$ subject to a given criterion. Mathematically, the function to be minimized can generaly be put into the following form,

$$H(o) = ||F_m - F||^2 + \alpha R(o) \tag{1}$$

where, F_m is the measured frequency response, and F is the Fourier transform of the estimate function o(r); $R(o)$ is the so called regularization function needed to ensure that the reconstructed o(r) has certain smoothness properties, and α is the so called regularization parameter that adjusts the degree of fitness expressed in the first term on the right hand side of (1) relative to the degree of regularization or smoothness expressed in the second term. For example, the function $R(o)$ in Tikhonov's regularization method[1] is taken to be a sum of the squared derivatives of o(r),

$$R_T(o) = \sum_k [o^{(k)}(r)]^2 \tag{2}$$

to ensure that o(r) has the required degree of smoothness. Here $o^{(k)}$ represents the k-th derivative of the function $o(r)$.

There are limitations however to all existing reconstruction algorithms; either an algorithm works well only for certain class of object functions or the *a priori* knowledge requirement is too stringent to be satisfied. The maximum entropy algorithm[4], which works well for point-like object functions, can be placed into the former class and the Papoulis-Gerchberg's algorithm[5],[6], which requires knowledge of the exact extent of objects, can be placed into the later class. By inspecting equation (1), one appreciates that reconstructions will be dependent upon the regularization function $R(o)$ chosen and a given $R(o)$ will only ensure a certain regularization (or smoothing) properties for the object function $o(r)$ and this is the reason why different algorithms with different $R(o)$ work well only for a certain set of object functions. For example, maximum entropy algorithm works well, as stated earlier, for point-like object functions and Tikhonov's regularization[1] is good for continuous object functions. This represents one difficulty of how to choose the regularization function

$R(o)$ in setting up the cost function $H(o)$ in (1). Another difficulty is how to choose the regularization parameter α for a given reconstruction problem. For practical reconstructions from noise contaminated data, the parameter α can be chosen mathematically depending on the signal-to-noise ratio in the data and this in turn introduces the added problem of having to estimate the signal-to-noise ratio which in practice is not easy to do.

Neural net models offer a new dynamical approach to collective nonlinear signal processing that is robust and fault tolerant and can be extremely fast when parallel processing techniques are utilized[3],[7]. Neural net models provide a new way of looking at signal processing problems that could offer solutions not thought of otherwise. A neural net processor for solving image reconstruction problems through minimization of an energy function of the type given in (1) has been studied earlier[3]. Here a neural net approach to the problem involving self-organization and learning is investigated. We will make use of the neural paradigm in a highly simplified and loose sense. Thus our nets allow for complex neurons and complex interconnection weights in addition to the more biological plausible real neurons and real interconnects. An adaptive three-layer neural net will be used to solve image reconstruction problems and learning is carried out in the net to change the interconnections between neurons in different layers by using the error back-propagation algorithm[8]-[11]. The analogy and relationship between the role played by hidden neurons and that played by regularization functions in neuromorphic solution of the image reconstruction problem in (1) will be discussed and it will be shown that hidden neurons play certain regularization role and that regularization functions in neuromorphic processors can be realized with hidden neurons. Learning in the approach presented here is shown to enable the neural net to form the regularization function $R(o)$ and the regularization parameter α automatically and to carry out near perfect reconstructions adaptively and with excellent robustness. The near perfect reconstruction results motivates the study of object recognitions with label representations and a three-layer nonlinear net will be discussed for practical radar target identification. A novel approach to achieve perfect (100% correct) identification of three test targets utilizing realistic data collected in an anechoic chamber environment using scale models of actual targets will also be presented. The findings support and demonstrate further the viability of neuromorphic automated target identification first proposed by Farhat *et al.*[16] as replacement to the traditional but considerably less economical approach of radar imaging.

2 Problem Formulation

For a spatially limited object function $o(r)$ and its Fourier transform $F(p)$, there exist the following well known relationships,

$$F(p) = \int_{-\infty}^{+\infty} o(r)e^{-jpr}dr \tag{3}$$

$$o(r) = \frac{1}{2\pi}\int_{-\infty}^{+\infty} F(p)e^{jpr}dp \tag{4}$$

$$\tag{5}$$

where the spatially limited $o(r)$ satisfies,

$$o(r) = \begin{cases} \neq 0 & \text{if } r \in [r_1, r_2] \\ = 0 & \text{otherwise} \end{cases} \tag{6}$$

The spatial frequency variable p here has unit of inverse length $[\mathrm{m}^{-1}]$ and the spatial frequency band corresponding to the frequency band $[\omega_1, \omega_2]$ used for measurement will be denoted as $[p_1, p_2]$. When the frequency response $F(p)$ is measured at equally spaced discrete frequency points over the measurement band $[p_1, p_2]$, that is at the frequency points,

$$p_k = p_1 + (k \triangleq 1)\Delta p \qquad k = 1, 2, \cdots, N \tag{7}$$

where N is the total number of measurements taken, and $\Delta p = (p_2 - p_1)/(N-1)$, the estimate of the object function by the discrete form of (4), i.e. the Discrete Fourier Transform (DFT) algorithm can be expressed as,

$$\begin{aligned} o(i) \triangleq o(r_i) &= \frac{\Delta p}{2\pi}\sum_k F(k)e^{jp_k r_i} \\ &= \frac{\Delta p}{2\pi}\sum_k F(k)e^{j[p_1+(k-1)\Delta p][r_1+(i-1)\Delta r]} \\ & \qquad i = 1, 2, \cdots, M \end{aligned} \tag{8}$$

where $\Delta r = (r_2 - r_1)/(M-1)$ is the object function sampling interval and M the total number of samples in the object domain. The resolution of the DFT estimation is known to be proportional to $2\pi/(p_2 - p_1)$ and it is not sufficient to discern object detail with spacing finer than $2\pi/(p_2 - p_1)$. Several methods for exceeding this resolution limit and achieving super-resolution have been studied in the past[4]-[6]. The limitation of these methods have been briefly mentioned in the introduction. Reconstructing microwave radar images from data processed by minimizing an energy function of the form given in (1)

through neuromorphic processing has been considered in earlier work[3]. Results of our continuing work on the relationship between the role of hidden neurons and regularization functions discussed in [3] are presented in the next two sections.

3 Neuromorphic Image Reconstruction

In this section we present a brief review of radar image reconstruction by neuromorphic processing[3] needed for subsequent discussion of the relation between the role of hidden neurons in layered nets and regularization functions. The function to be minimized in microwave radar imaging by neural net processing[3] has the same form as that in (1),

$$H(o) = ||F_m - F||^2 + \alpha R(o) \tag{9}$$

All quantities in (9) are the same as defined earlier and the norm defined in the complex space C is of the following form,

$$||F_m - F||^2 = \sum_{i=1}^{N} |F_m(i) - F(i)|^2 \tag{10}$$

When the Fourier transform F is expressed in terms of the object function o(r), the energy function $H(o)$ in (9) will only be a function of the variable o(r), since F_m is the measured frequency response and is known. After some manipulations and by assuming the object function to be reconstructed in microwave radar imaging is real (see ref. [3] for details), the following state update equation for the neuromorphic processor can be obtained,

$$o^{(j+1)}(k) = o^{(j)}(k) + \Delta o(k) + \lambda I_k \qquad 0 < k \leq M \tag{11}$$

$$\Delta o(k) = \lambda \left[2 \sum_{i=1}^{M} \Re[T_{ki}] o^{(j)}(i) - S_k \right] \tag{12}$$

where $o^{(j)}(k)$ represents the state of the k^{th} neuron at the j^{th} iteration; λ is defined as the gain of the k^{th} neuron; T_{ki} is a quantity which bears information about the transformation (here the Fourier transform) from the space $\Theta \ni o$ to the space $\Omega \ni F$ and the term I_k represents the available information F_m and is given by,

$$I_k = 2\Re \left[\sum_{i=1}^{N} F_m(i) K_{ik}^* \right] \tag{13}$$

where $K_{ik} = c \cdot e^{jp_i r_k}$ represents the Fourier kernel and c is a constant. Equation (13) is identified as the real part of the complex object function generated by Fourier inversion of

the measured frequency response F_m. The term S_k in (12) is viewed as a regularization related adaptive threshold and is given by the following expression with A_{kk}, $A_{k(k-1)}$, and $A_{k(k+1)}$ being given constants[3],

$$S_k = 2\alpha[A_{kk}o^{(j)}(k) + A_{k(k-1)}o^{(j)}(k-1) + A_{k(k+1)}o^{(j)}(k)] \tag{14}$$

for a stabilizing (regularizing) function of the following form in Tikhonov's regularization method,

$$R(o) = \sum_{i=1}^{M} \left[o^2(i) + \left(\frac{o(i) - o(i-1)}{\Delta r} \right)^2 \right] \Delta r \tag{15}$$

or in its equivalent continuous form,

$$R(o) = \int \left\{ o^2 + [o'(r)]^2 \right\} dr. \tag{16}$$

The neural net update transformation as expressed in (11) is carried out iteratively until the global minimum of the energy function of (9) is reached.

Microwave radar images reconstructed (see [3] for details of the tomographic reconstruction method) using the neural net processor described in (11) showed improvement over images reconstructed by DFT algorithm when Tikhonov's stabilizing function in (15), or equivalently an adaptive threshold linearly related to the neural states as expressed in (14), was used[3]. In conventional neural nets, binary neurons are used and nonlinear mapping of neural states is used[7] and that is largely responsible for the robust and fault tolerant collective signal processing properties of neural nets. The neural state update equation in (11) is a linear iterative equation when the threshold of linear mapping of neural states given in (14) is used; in this case, the advantage exploited in a neural net using (11) to solve the problem in (9) is only the parallel processing capability of the neural net. No use is made of nonlinear mapping. For the problem of image reconstruction in (9), multi-valued (analog) neural states have to be used to represent the bipolar object function. Therefore, in order to make the neural net processor in (11) more neuromorphic, nonlinear mapping can be introduced only via the adaptive threshold S_k. A nonlinear function of the form,

$$g(S_o) = \tanh(S_o) \tag{17}$$

similar to the sigmoidal function widely used in conventional neural nets[7],[8] was introduced heuristically and employed for the adaptive threshold with S_o being a linear combination of the neural states[3]. The adaptive threshold S_k in (14) is a linear combination

of three nearest states only and S_o in (17) denotes a linear combination of possibly many states in general. The neural state update equation in (11) can then be written as,

$$o^{(j+1)}(k) = o^{(j)}(k) + \Delta o(k) + \lambda I_k \qquad 0 < k \leq M \tag{18}$$

$$\Delta o(k) = \lambda \left[2 \sum_{i=1}^{M} \Re[T_{ki}] o^{(j)}(i) - g(S_o) \right] \tag{19}$$

The neural net processor in (18) was used to reconstruct 1-dimensional functions (range-profiles) from measured frequency response data F_m for a sufficiently wide range of aspect angles of a scaled model of an aerospace test object and a 2-dimensional object function representing a projection image of the test object was formed by coherently summing the back-projections of the 1-dimensional range-profiles based on the projection-slice theorem[3],[12]. For details of this reconstruction the reader is referred to [3]. The scale model used is that of a B-52 airplane and realistic frequency response data F_m for it were gathered for a range of aspect angles in an anechoic chamber microwave scatter measurement facility for two different frequency bands: one extending from 6(GHz) to 17(GHz) and the other from 2(GHz) to 26.5(GHz). Images reconstructed from the two frequency bands by DFT inversion and back-projection are shown in Fig. 1(a) and (b), respectively. The image in Fig. 1(b) from the wider frequency band of 2(GHz) to 26.5(GHz) has higher resolution as would theoretically be expected. It clearly shows the double barreled nature of the engines which is not clearly delineated in the image in Fig. 1(a). The image reconstructed from frequency response data acquired over the narrower band (6(GHz) to 17(GHz)) using the neural net processor expressed in (18) with the nonlinear threshold mapping function (17) is shown in Fig. 1(c); this image has almost the same resolution as the image reconstructed over the band from 2(GHz) to 26.5(GHz) and the double barreled nature of engine is clearly delineated in the image. The image quality obtained from the neural net processor expressed in (11) with linear threshold mapping function was inferior to that in Fig. 1(c) demonstrating the importance of incorporating nonlinearity[3]. These results demonstrate the high resolution capability of nonlinear neural net in image reconstructions.

4 Relationship Between the Role of Hidden Neurons and Regularization Functions

The neural net processor expressed in (18) is basically of the Hopfield variety[7]. It works iteratively until a stable state of the net is reached to give a solution for the image recon-

struction problem of (9). The iterative process can be implemented by a parallel feedback loop[3] in which the new state of the net for the update iteration is obtained by feedback of the state change $\Delta o(k)$ computed from the neural state for the preceding iteration and this is schematically illustrated in Fig. 2(a). The computation of $\Delta o(k)$ can be implemented by a subnet with one hidden layer of neurons as shown in Fig. 2(b). By comparing (18) with Fig. 2(b) it is noted that the hidden layer neurons implement the nonlinear adaptive threshold related to the regularization function. The point is that the weights (or synaptic connections) used for the adaptive threshold can be combined with other weights which directly connect the input layer with the output layer if the adaptive threshold is a linear mapping of the neural states like that shown in (14) and in this case the neural net update equation, (11) for example, can be rewritten as,

$$o^{(j+1)}(k) = o^{(j)}(k) + \Delta o(k) + \lambda I_k \qquad 0 < k \leq M \tag{20}$$

$$\Delta o(k) = 2\lambda \sum_{i=1}^{M} \left[\Re[T_{ki}] - \alpha\delta_{kk}A_{ki} - \alpha\delta_{(k-1)i}A_{ki} - \alpha\delta_{(k+1)i}A_{ki} \right] o^{(j)}(i) \tag{21}$$

where $\delta_{i,j}$ is the Dirac delta function. On the other hand, the total connections implemented from input layer through hidden layer to output layer in Fig. 2(b) can not be combined with other direct connection weights from input layer to output layer. This shows the necessity of implementing the adaptive threshold representing a regularization function in nonlinear neural nets with a hidden neural layer.

The relationship between the role of hidden neurons and regularization functions can also be appreciated by examining the regularization role played by hidden neurons. Hidden neurons are used to generate internal representations in neural networks and to extend the computational (or mapping) power of simple two-layer associative networks[8]. In simple two-layer associative networks, input patters at the input layer are directly transformed (or mapped), through the synaptic connections between neurons, into output patterns at the output layer and there is no internal representations by hidden neurons involved in such a network. Because of this direct mapping property, simple networks will transform input patterns of similar structure into output patterns of similar structure; consequently, such network will not be able to give desired mapping outputs which are quite different (or similar) when the inputs are quite similar (or different). A classic example of this situation, that has been discussed by other researchers[8], is the exclusive-or (XOR) problem illustrated in table 1.

In this example, the inputs (for example, 00 and 11) which are quite different are desired

input pattern	output pattern
00	0
01	1
10	1
11	0

Table 1: XOR Mapping

to be mapped into the same output (for example, 0). If two neurons in the input-layer are used to represent the two input bits and one neuron in the output-layer is used to represent the one output bit in a simple two-layer network, it is impossible to find a set of weights and thresholds for all the neurons in such a network to perform the desired mapping[13]. The difficulty for a simple two-layer net without hidden neurons in solving the XOR mapping problem lies in mapping quite different patterns (11 and 00) to identical output (0) as well as mapping quite similar patterns (01 and 10) into identical output (1). This pair of mappings are quite contradictory and by concepts and definition of ill-posedness this kind of mapping is an ill-posed mapping. For example, in inverse scattering, the mapping (inverse) is known to be ill-posed if the solution of the mapping or reconstruction does not exist or is sensitive to noise in the input data. In the XOR problem in a two-layer neural net, a network to perform the mapping cannot be found and this can be interpreted as similar to ill-posed problems as no solution for the problem exist.

On the other hand, a layer of hidden neurons inserted in-between the input layer and the output layer of a simple two-layer network will enable the network to perform arbitrary mapping from input to output via the hidden neurons if an adequate number of hidden neurons is utilized[8],[13]. It can be easily verified that the network with a single hidden neuron shown in Fig. 3 can perform the XOR mapping mentioned above. The network shown in Fig. 3 overcomes the difficulty encounted in a 2-layer net by using a hidden neuron to change the quite different input patterns into patterns with *sufficient similarity* as seen by the output-layer neuron; it accomplishes the task by using one hidden neuron for a two-bits to one-bit mapping as detailed in Fig. 3. The numbers on the arrowed lines represent the required weights of synaptic connections among the neurons and these are ultimately determined through learning (see for example [8]-[11]). The numbers in the

circles represent the required thresholds of the neurons and they have been assumed to be fixed beforehand in the example shown here. All the neurons in the net are assumed to have only two states: on (1) or off (0). The hidden neuron has output 1 (on) only when both input neurons have states 1 (on) and it has output 0 (off) otherwise; The output neuron will be turned on (1) when it has a net positive input greater than 0.5, and the output neuron will be turned off (net input smaller than 0.5) by the hidden neuron output through the synaptic connection weight of -3.0 when both input neurons are on (1). From the point of view of the output neuron, the inputs to it are quite similar when either the input neurons are on (11) or off (00). Thus, the role of regularization or constraint function played by the hidden neuron in this case is to change the degree of similarity among the input patterns corresponding to the same output pattern. This role of providing additional constraints among input neurons by hidden neurons can be considered to be the same as that of regularization functions for ill-posed problems.

The regularization role played by hidden neurons can also be appreciated from the error back-propagation (EBP) algorithm in which hidden neurons are used[8]-[11]. The EBP algorithm for a general problem is also formulated so as to minimize the error energy function,

$$E = ||O - \tilde{O}||^2 \tag{22}$$

where, $\tilde{O}$ is the specified or the desired output and O is the output of the network for a given input. For the given input and the specified output, the error signal given by E is fed-back (or back-propagated) into the network to adjust the interaction weights (weights of synaptic connections) among all neurons including hidden neurons. This, so called learning procedure, is iterated until a set of weights is arrived at for which the specified output or equivalently the specified minimum of the energy function is reached. Comparison of the energy function in (9) with that in (22) shows there is no regularization operator involved in (22). It is well known that inversions by minimizing the error energy function of the form shown in (22) in the presence of noise are ill-posed and outputs are usually not stable with respect to inputs. From our study of networks with hidden neurons, it is found that the performance of the networks is quite robust with respect to inputs as shown by simulation results presented next. The role played by the regularization operator in (9) to constrain the output in ill-posed mapping problems is achieved with the hidden neurons in neural networks. Impossible mappings in a neural network can be made possible by increasing

the number of hidden neurons and this can be explained by the fact that regularization is introduced or further enforced by the increase in number of hidden neurons.

5 Reconstruction by Neural Nets Through Learning

The iterative neural net equation (11) can be cast in a closed form of a non-iterative equation and implemented with a non-iterative processor when an adaptive threshold that is a linear function of neural states, (14) for instance, is used. On the other hand, when the adaptive threshold that is a non-linear function of neural states of (17) is used, the iterative neural net equation (18) can not be written in the closed form of a non-iterative equation and there is no known method to directly implement the iterative equation with a non-iterative processor; this results from the difficulty of choosing different regularization $R(o)$ and different parameter α in (9) for different reconstruction problems, since the first term on the right hand side of (9) can be computed with a non-iterative DFT processor. This difficulty can be overcome by a neural net through learning which enables forming $R(o)$ and α automatically depending on the image to be reconstructed as will be clarified below.

Hidden neurons have been shown to have regularization effect and hence a hidden neural layer will be used here for the purpose of regularization to overcome the ill-posedness of image reconstruction from partial frequency response. A three-layer neural net with feed-forward connections for image reconstruction is schematically shown in Fig. 4. The input layer takes the frequency responses from measurements and neurons in the input layer, which are complex (i.e. their states are complex and equal to the real and imaginary values of the measured complex frequency response), are connected to neurons in both the output layer and the hidden layer. The synaptic connection from neurons in the input layer to neurons in either the output layer or the hidden layer are complex and will be fixed and taken as the Fourier weights for the image reconstruction problems in situations in which the measurement data and the image to be reconstructed have a Fourier transform relation. The number of neurons in all three layers will be taken to be the same for the moment and equal to the number of frequency points at which the response is measured. Images to be reconstructed are assumed to be normalized to unity and the output from neurons in the hidden layer will take a nonlinear function of the form $\tanh(\cdot)$. Mathematically, the final

output neural states representing the image to be reconstructed is,

$$o(i) = z(i) + \tanh\left[\sum_j r_{ij} z(j)\right] \tag{23}$$

where r_{ij} is real valued synaptic link between ith neuron in the output layer and jth neuron in the middle (hidden) layer and,

$$z(l) = \Re\left[\sum_{k=1}^{N} W_{lk} F_m(k)\right] \quad l = i, j \tag{24}$$

where $\Re[\cdot]$ represents real part of the bracketed quantity and W_{lk} are the Fourier weights. Once more a real object function $o(i)$ has been assumed for microwave radar imaging[3] and $z(l)$ is recognized as the real part of Fourier inversion of the measured frequency data F_m. Learning in the neural net will be carried out next and it involves determining the synaptic weights r_{ij} by an error back-propagation algorithm[8]-[11].

With an error back-propagation algorithm, the neural network can be made to learn under supervision to perform extrapolation and reconstructions as follows: for a given desired or ideal object function D, when the measured frequency response $F_m(p)$ is fed into the network in Fig. 4 and the output from the network denoted as o, an error function,

$$E = ||D - o||^2 = \frac{1}{2}\sum_i |D(i) - o(i)|^2 \tag{25}$$

can be defined. Since knowledge of the desired object function D at the output of the net is required (D is also the ideal desired image at the output), the learning is supervised. Using the chain differentiation rule, the change of the error function with respect to the change of weight r_{ij} can be written as,

$$\frac{\partial E}{\partial r_{ij}} = \frac{\partial E}{\partial o(i)} \cdot \frac{\partial o(i)}{\partial r_{ij}} \tag{26}$$

From equation (25),

$$\frac{\partial E}{\partial o(i)} = -[D(i) - o(i)] = -\delta_i \tag{27}$$

and from equation (23),

$$\begin{aligned} \frac{\partial o(i)}{\partial r_{ij}} &= \tanh'\left[\sum_j r_{ij} z(j)\right] z(j) \\ &= z(j)/\cosh^2\left[\sum_j r_{ij} z(j)\right] \end{aligned} \tag{28}$$

Combining now (26), (27) and (28), the following equation is obtained,

$$\frac{\partial E}{\partial r_{ij}} = -\delta_i z(j)/\cosh^2\left[\sum_j r_{ij} z(j)\right] . \tag{29}$$

To reduce the error signal in (25), the weight r_{ij} can be changed through gradient descent by an amount,

$$\begin{aligned} \Delta r_{ij} &= \eta\left(-\frac{\partial E}{\partial r_{ij}}\right) \\ &= \eta\delta_i z(j)/\cosh^2\left[\sum_j r_{ij} z(j)\right] \end{aligned} \tag{30}$$

with η being an constant controlling the learning rate.

The above procedure is for one given object (or pattern) function D. When there are M ideal images of interest, the procedure is carried out M times, once for each image. For each image the error signal is checked and if a specified error criterion (to be specified below) is not satisfied, the procedure is repeated again for every pattern; this is repeatedly done until the error signal criterion is satisfied for each image.

6 Simulation Results and Robustness Tests

Simulations were carried out to verify the learning concept presented above. Several ideal object functions of spatial extent within [0,4](cm) are used. The number of neurons for the input, middle, and output layers are assumed to be the same and equal to 21 neurons for each layer. The small number of neurons used and the small extent [0,4](cm) of the function occupied are all chosen for the purpose of containing the computations involved but they can be increased or altered at will to any desired values. The frequency response of the object function chosen is synthesized (computed digitally) in the (6-17)[GHz] range and subjected in simulation to the action of the network in Fig. 4. The network can determine a set of r_{ij} links for a given set of functions to produce correct patterns within the specified error criterion. The error criterion used is $\max_i |D(i) - o(i)| < 0.097$.

One of the simulations has been done for a set of two object functions: the first one is,

$$o_1(r) = \begin{cases} 1.0 & r \in [0.2, 1.2](\text{cm}) \\ 0 & r \in [0, 0.2)(\text{cm}) \text{ or } r \in (1.2, 4.0](\text{cm}) \end{cases} \tag{31}$$

and the second one is,

$$o_2(r) = \begin{cases} 1.0 & r \in [2.2, 3.2](\text{cm}) \\ 0 & r \in [0, 2.2)(\text{cm}) \text{ or } r \in (3.2, 4.0](\text{cm}) \end{cases} \tag{32}$$

These two function are shown in Fig. 5(a) and (b), respectively, and their spatial extents are seen to be within $[0, 4](\text{cm})$. The frequency responses of the two object functions synthesized over the frequency window $[6, 17](\text{GHz})$ are shown in Fig. 6(a) and (b), respectively. If the DFT inversion method is applied to the frequency data in Fig. 6, a low resolution image with most its intensity concentrated around the sharp edge of the object functions will be obtained, because the frequency information in Fig. 6 is over a relatively high frequency window. Shown in Fig. 7 is the reconstruction of the first object function from the partial frequency domain data in Fig. 6(a) by the DFT method and it is seen that there is a relatively broad positive pulse at the position of the rising edge of the original object function and a broad negative pulse at the position of the falling edge of the original object function; the two pulses are also of different amplitude, although the given object function has the same rising edge and falling edge. When the two object functions are alternately presented to the network in Fig. 4 and the synaptic connections are changed according to (30) of the learning algorithm discussed in section 5, the learning process gradually converges and a set of synaptic connections is learned by the network to give near perfect reconstructions of the object functions within the specified error criterion when the frequency response of either object function is presented to the network. The network accomplishes the learning in just five learning cycles and a learning cycle is defined as the whole process of presenting once the two patterns to the network and modifying the weights following each pattern presentation. The value of η used was 0.99. We will discuss the choice of η further below. Figure 8 shows the outputs of the network for several typical learning cycles and demonstrates how the network gradually learns the two patterns by adjusting its connection weights. Shown in Fig. 8(a) are the outputs of the network for the first pattern (on the left side of Fig. 8(a)) and for the second pattern (on the right side of Fig. 8(a)) after the network has been trained with the first pattern only during the first learning cycle. It is seen from Fig. 8(a) that the output from the network for the first pattern as input is near perfect and the output for the second pattern as input is not like the second pattern at all but resembles more the first pattern; this is understandable, since the network has only learned the first pattern and it has not seen the second pattern yet. Completing the first learning cycle by training the net next with the second pattern, we find the network is able to give near

perfect reconstruction for the second pattern as input shown on the right side of Fig. 8(b) but the output when the first pattern is presented is seen to have been altered and is not as good as before, being much more like a superposition of the first pattern and the second pattern. This may be interpreted as the network having lost some of its previous internal representation of the first pattern during the learning of the second pattern. The internal representation of the first pattern is restored however in the second learning cycle following the presentation of the first pattern again to the net (left side of Fig. 8(c)). The output (on the left side of Fig. 8(c)) from the network for the first pattern as input approaches now again a near perfect reconstruction and the output (on the right side of Fig. 8(c)) for the second pattern as input is much better than that previously obtained on the right side of Fig. 8(a) during the first learning cycle. This result is also understandable since so far the network has been trained with the first pattern twice (during the first learning cycle and the second learning cycle) and with the second pattern once only (during the first learning cycle). The output (on the right side of Fig. 8(d)) for the second pattern is improved during the second learning cycle after presenting the second pattern to the network for learning and this degrades the performance (showns on the left side of Fig. 8(d)) of the network in recognizing the first pattern. By repeatedly and alternately presenting the two patterns to the network for learning, it gradually adjusts its interconnection weights to improve the reconstructions for both patterns. Shown in Fig. 8(e) and (f) are the outputs of the network during the third learning cycle after the first pattern and the second pattern have been presented to the network, respectively; the performance of the network is seen to have improved in comparison with the corresponding cases in the second learning cycle. After the first pattern has been presented to the network for learning during the fourth learning cycle, the outputs shown in Fig. 8(g) for both patterns are much better except for the presence of some side lobes in the output shown on the right side of Fig. 8(g) for the second pattern as input. The side lobe level is reduced to the specified tolerable error range of $\max_i |D(i) - o(i)| < 0.097$ during the fifth learning cycle as shown in Fig. 8(h) where the outputs of the network are given for both patterns after the network has been presented with the first pattern for learning during the fifth or the final learning cycle.

How to choose the learning rate η is critical to the speed of learning process and the range of suitable learning rates can be analytically determined for learning algorithms involving a linear function of neural states[14]. For learning algorithm involving a nonlinear function of neural states given in (30), it is however hard to analytically determine the range of

the learning rate. By inspecting (30), it is seen that the learning rate η represents the proportion by which the synaptic weight changes in accordance to the output error induced by the current synaptic weights themselves. In our preceding simulations, the learning rate is usually chosen as $\eta = 0.99$. It would not make sense to have the learning rate η greater than 1 as indicated elsewhere[14], since making the learning rate greater than 1 could overcorrect the output error and this has been observed in our simulations. What is meant by overcorrection here is that the output energy error which we seek to minimize exhibits oscillations and sometimes is increased. Overcorrection usually results in longer converging time. On the other hand, making the learning rate too small could also slow down the learning process. Another cautionary remark in carrying out the learning process is that the initial synaptic weights should not be equal; otherwise, the network would obtain identical weights for all synaptic connections and this has also been noticed in other studies[8]. The initial synaptic weights in our study were chosen randomly.

More complex shaped object functions have also been used to test the learning and reconstruction capability of the neural net in Fig. 4. A set of two object functions is used and these are shown in Fig. 9. The first function in Fig. 9(a) has a spatial extent $[0.2, 0.8]$(cm) and is similar to that shown in Fig. 5(a). The second function is of more complicated shape and the first part of this function is a pulse of width 0.8(cm) and the second part is of triangular shape. After a set of synaptic weights is learned by the network when the two patterns have been presented to the net only five times using the learning algorithm discussed in section 5, the network were able to give a near perfect reconstruction when the frequency response of either function is presented to it. The reconstructions of the two object functions by the network are shown in Fig. 10. By comparing Fig. 10(b) and Fig. 9(b), it is seen that the reconstruction of the triangular portion of the second object function is perfect; since the triangular part of the second function resembles more the undulations of a continuous function, its perfect reconstruction appears to imply the network performs better for continuous functions.

Generaliztions and Robustness: The two simulations presented above have shown good results when the network is used for reconstructions of object functions which it has been presented with during learning process. Generalization in neural networks is an issue of practical importance[14]. It deals with the performance of a network when inputs are not specifically among the training sets the net has been presented with during the learning process but are similar to them. Generalization in the network in Fig. 4 for extrapolations

and reconstructions from partial frequency information is studied here from the point of view of the network's performance with noise contaminated frequency response input data.

From the discussion in section 4, it can be appreciated that hidden neurons play certain regularization role and it is the regularization that makes the solution stable for problems of extrapolations and reconstructions from partial frequency informations. The network with hidden neurons in Fig. 4 provides the regularization needed and is expected to give stable and robust reconstructions even in the presence of noise. Numerical simulations were carried out to verify that. One of the simulations was done with the test object functions used previously and shown in Fig. 5. The frequency responses of the two object functions in Fig. 5 were contaminated with Gaussian noise with the following distribution function,

$$G(N) = \frac{1}{\sqrt{2\pi}\sigma} e^{-N^2/(2\sigma^2)} \tag{33}$$

where N represents the noise amplitude, and σ^2 is the variance of Gaussian noise. Defining the signal-to-noise ratio (SNR) as,

$$\begin{aligned} \text{SNR} &= \frac{\text{average signal energy in the given frequency band}}{\text{noise variance}} \\ &= \frac{1}{p_2 - p_1} \int_{p_1}^{p_2} |F(p)|^2 dp/\sigma^2 \end{aligned} \tag{34}$$

we find when SNR=5, the noise contaminated frequency responses for the two object functions are as shown in Fig. 11 for the frequency band [6,17](GHz) corresponding to $p \in [2.5, 7.1](\text{cm}^{-1})$. The difference before and after noise contamination can be seen by comparing Fig. 6 and Fig. 11. Even though the frequency responses in Fig. 11 after noise contamination differs appreciably from the noise free frequency responses in Fig. 6, the network, which learned a set of synaptic connections when the noise free frequency information is used in the learning process, is still able to give the very good reconstructions shown in Fig. 12 when the noise contaminated frequency information is presented to it. The reconstructions in Fig. 12 from the noise contaminated frequency information show weak side lobe structure compared with the reconstructions in Fig. 8(h) when noise free frequency information is used as input. When the SNR is further decreased, the side lobe structure in the reconstructions from noise contaminated frequency information will increase. The reconstruction from noisy frequency response data can be improved by training the network with noise free frequency data as well as some noise contaminated frequency data. For studies with the two test patterns considered here, the network was trained with noise free frequency data shown in Fig. 6 and also with noisy frequency responses (SNR=1) shown in

Fig. 13. The ideal patterns needed in the supervised learning process for the noise free and the noisy data were specified to be the same as those shown in Fig. 5. The noise free data and the noisy data were presented alternately to the net to adjust the connection weights until the specified error criterion $\max_i |D(i) - o(i)| < 0.097$ for every pattern was reached. When the resulting network is tested with the noisy frequency response data shown in Fig. 11 of SNR=5 presented as the input after the stated training, the outputs from the network are as shown in Fig. 14. In comparing of Fig. 14 with Fig. 12, the improvement of the side-lobe structure in Fig. 14 can be clearly seen as result of mixing instances of noisy and noise-free data sets in training the network. In practice, a network, being trained with examples of data from its environment, is expected to encounter such examples at differing levels of SNR. The findings above suggest that this could be beneficial for enhancing performance of the net.

7 Radar Target Identification by Layered Network

From the preceding discussion, it is seen that robust extrapolation and near perfect reconstruction can be achieved with layered nonlinear networks. An interesting issue is whether there always exists a network which can do extrapolations and reconstructions for a given finite number of functions or patterns of interet. A theorem[8],[15] concerning multi-layer neural networks, which simply states that a multi-layer network with sufficier t number of hidden neurons is able to perform any kind of mapping from input to output, makes it possible for the network shcwn in Fig. 4 to perform extrapolations and reconstructions of any finite number of functions of interest, if enough of hidden neurons are used in the network. For a finite number of aerospace targets, a 2-dimensional object function describing the geometrical shape of each target can be formed as discussed in [3] from the 1-dimensional functions reconstructed by a learning net, as described in the last section, through extrapolation of partial frequency response data acquired for fixed aspects of the targets over a sufficiently wide range of aspect angles. The 2-dimensional image obtained in this fashion can provide high enough resolution through data acquisitions over a wide range of aspect angles and extrapolations of the measured frequency response data for every aspect. The high resolution image, like those shown in Fig. 1, would enable a human observer to recognize and identify the target. Another more attractive and less involved concept in target identification does not involve forming an image. It provides for target

identification from an identifying label of the target generated by a neural net automatically from input information (i.e. frequency response data) belonging to that target[16]. This approach is necessary in situations where aspect information (frequency response echos for various aspect) of the target can not be obtained over a sufficiently wide range of aspect angles because of practical limitations and consequently a high-resolution image of the target can not be formed[16]. The issue then is that of radar target identification from a single frequency response echo for any practical aspect of the target, or a few such echos, by a layered nonlinear network through self-organization and learning as will be discussed in this section.

The traditional approach in nonimaging radar target recognition has been to extract from suitably formed radar echos characteristic features or signatures of the targets and to compare these with a library of such signatures[17]. This kind of approach is basically a parametric estimation method and makes certain assumption about the form of the return signals or echos as expressed by several parameters. The extraction of the assumed parameters used in the approach is usually sensitive to noise[18] and there is no adaptation involved.

The network used for target recognition in our work is shown in Fig. 15. This network is a variation of the network shown in Fig. 4 used for extrapolations and reconstructions. The network of Fig. 4 has been shown to be robust in extrapolation and reconstruction from partial information and the number of output neurons in the network was equal to the number of samples representing the function to be reconstructed. The network shown in Fig. 15 is intended to perform robust target recognition from partial information and the number of output neurons in the network is chosen now to allow forming enough distinguishable labels to represent all targets of interest. Using labels instead of object functions makes learning easier, since the ideal object functions, that are needed to accomplish learning for extrapolations and near perfect reconstructions and that are not easy to obtain for aerospace targets in general, are now not required. Since label representations rather than object functions of targets are to be used now for identification, no direct connections between output neurons and input neurons in Fig. 15 are used and this simplifies the structure of the network. The connections from input neurons to hidden neurons accomplish as before Fourier mapping as in the network of Fig. 4, i.e.,

$$z(j) = \Re[\sum_{k=1}^{N} W_{jk} F_m(k)] \tag{35}$$

where W_{jk} represents the Fourier weight for inverting the known (measured) partial frequency domain information $F_m(k)$. For target recognition from other than frequency domain information, the weights W_{jk} are set up in accordance to the specific transform applicable or could be determined also through training as for the connection weights from hidden neurons to output neurons to be discussed next and nonlinear mapping can also be introduced for the hidden neuron states. The input to an output neuron in Fig. 15 is given by,

$$u_i = \sum_j r_{ij} z(j) \tag{36}$$

where r_{ij} again represents the weight from neuron j in the hidden layer to neuron i in the output layer to be determined by learning. The output neuron state is now given by the expression,

$$o(i) = U[\tanh(u_i)] = \begin{cases} 1 & \text{for } u_i > 0 \\ 0 & \text{for } u_i < 0 \end{cases} \qquad i = 1, 2, \cdots, M \tag{37}$$

where $U[\cdot]$ is the unit step function and the form $U[\tanh(u_i)]$ is used in (37) to show more clearly the nonlinear summation input to the output layer and the evolution of the circuit in Fig. 15 from that of Fig. 4. Different targets are represented by different output state.

Two groups of test targets have been used in our study: the first group contains a 100 : 1 scale model of a B-52 aircraft and a 150 : 1 scale model of a Boeing 747 airplane model; the second group contains a 75 : 1 scale model of a space shuttle in addition to the two scale models in the first group. Sketches of all three scale models with their actual dimensions are shown in Fig. 16. It is noticed that the shapes of the Boeing 747 and the space shuttle are relatively less complex than that of the B-52 airplane. Since only three aerospace target models are used, two output neurons are used to provide label representations for three targets; two output neurons can usually provide labels for 2^2 (= 4) distinct patterns. The state (0,0) of the output neurons in the network shown in Fig. 15 is left idle to indicate the situation when there is no information input to the network.

To study radar target identification with practical application in mind, it would be necessary to examine the performance of the network for all possible aspects of the target that could be encountered in practice by the observer (the radar system) and this entails massive data collection and storage. Because of limitations of our experimental facility, frequency response data of the targets are collected for only a limited range of aspect angles

extending over a range of 20° in azimuth from head-on (0°) view of the targets to 20° towards the broad-side view of the targets. The elevation angle of the target was fixed at 15° relative to the horizontal. The results obtained with this limited data set are however quite telling and representative of what can be expected with larger libraries of frequency responses covering all target aspects of interest. Frequency domain data are collected for 100 aspect views equally spaced over the 20° for each target and this represents a separation of 0.2° between adjacent views. The results presented next show how correct recognition depends on the angular spacing between adjacent views and how perfect recognition from a single echo or look can be achieved with the network of Fig. 15.

The network was first presented with frequency response data belonging to certain percentage of the 100 aspect views of the targets for learning and each target is assigned a label: (0,1) for B-52, (1,0) for Boeing 747, and (1,1) for space shuttle. There are 101 frequency points collected over the band [6.5, 17.5](GHz) for each aspect view and the number of neurons in the input layer is also chosen to be 101 to represent the number of frequency samples for an individual aspect angle. The number of neurons in the hidden layer was chosen to be equal to the number of neurons in the input layer and also to be 101. For learning, the error back-propagation algorithm described in section 5 for the network of Fig. 4 also applies to the network shown in Fig. 15 and enables adjusting the connection weight r_{ij} between output neuron and hidden neuron. When the frequency response of a target for a specific aspect angle is presented to the network, the network iteratively adjusts the weight r_{ij} by error back-propagation until the desired label for the target is given by the network. The training data (frequency response for different aspects or views) are presented in turn to the network for each target and all targets of interest are learned by the network in turn. The process of presenting all the training data for all targets once constitutes one learning cycle. The maximum number of iterations observed for the network to learn a specific target of the types used in our study at the start of the learning process is 7 and the number of iterations decreases as learning progresses or the number of learning cycles increases. When the network has assimilated and learned the correct representations for all targets, the learning process is terminated. The maximum number of learning cycles observed for the network to learn all targets is 8. All figures mentioned next are aimed at illustrating the simple learning process of the network shown in Fig. 15 for recognitions of typical aerospace targets.

Shown in Fig. 17 is the performance of the network for the first group of targets, the

B-52 and the Boeing 747 scale models. It has been mentioned earlier that there are 100 aspect views (frequency responses) collected for each target. The curves in Fig. 17 show the probability of correct recognition by the network of the two targets versus the percentage of aspect views used for training. The percentage for training is taken with respect to the set of the total 100 aspect views collected. The training set can be selected deterministically, i.e. in a given order, or randomly from the set of 100 aspect views characterizing each target. The criterion for choosing the training set is to make sure that information about the target is evenly represented. Therefore, for example, the deterministic selection case of 50 percent of the available aspect views as the training set can be formed by selecting every other aspect view, that is, all the even (or odd) numbered views out of the total 100 available aspect views. For the random selection case, the training set can be formed by selecting aspect views out of the total angular window of 20° with even probability. Our study shows that the performance of the network when tested is virtually not affected by whether the training set is selected deterministically or randomly and that at most a 1% discrepancy in results for the two cases is observed. The test of the performance of the network after it has been trained is done with all aspect views collected. Thus a certain percentage of the test set would have been used in training the network and the remainder of the test set has never been seen by the network before. Even though the incremental spacing between viewing angles for the given set of test data is small (0.?°), it is seen that the network achieves percent correct recognitions of only 54% for the B-52 airplane and 72% for the Boeing 747, when 10% of the total number of available views was used in the training process or equivalently when the views with roughly 2° angular separation have been used for training. The performance of the network improves nonlinearly as the percentage of views used for training is increased. The network is seen to perform much better in recognizing the Boeing 747 than in recognizing B-52 and this is because the shape of Boeing 747 is less complex than that of B-52 and this enables the network to capture the underlying structure of the Boeing 747 in its internal representation (the r_{ij} weights) much faster than for the B-52. The percent correct recognition of the network reaches 90% when the percentage of views used for training increases to 40% for the B-52 and 20% for the Boeing 747, respectively. When the percentage of views used for training for both targets increases to 60%, the network can recognize more than 98% of the testing aspect views presented to it correctly.

For the network shown in Fig. 15, with the connection weights from input layer neurons

to the hidden layer neurons fixed as the Fourier weights, the input to the hidden layer can be interpreted as the real part of the Fourier inverse of the measured frequency response data F_m for one aspect view as mentioned earlier and this input, termed range-profile, to the hidden layer bears information, such as rough extent, shape, fine structure, etc., of the target as seen from that aspect angle[3]. What the network accomplishes during training is to extract common features or certain correlations from the training data to form a representation for the target by adjusting its weight r_{ij}. When the network is tested with test views, the portion of the test views which have not been presented to the network during training can be considered as noisy versions or "correlates" of the training set. This ability of the net to generalize, i.e. to recognize noisy data or correlated data is an attractive feature of neuromorphic signal processing. The range-profile data in various aspect views of a complex aerospace target can differ noticeably from one aspect angle to another and the curves in Fig. 17 show that for a required correct recognition performance, of 90% for example, the percentage of views, 40% for the B-52 and 20% for the Boeing 747, or a minimum corresponding angle spacings between adjacent views used in the training sets, approximately 0.5° for the B-52 and 1° for the Boeing 747, respectively, should be used for training to achieve the 90% level of recognition. In fact, since the data in various aspect views for complex shaped aerospace targets change markedly from one aspect angle to another, the resemblance or correlation of adjacent views for some aspect angles are so weak even for the angular spacing of 0.2° used in out data acquisition that the network fails to recognize the targets perfectly (with 100% score) even if almost all views have been used for training; this is evident in Fig. 17 when correct recognition for both targets did not reach 100% before 100% of the available aspect view data have been used for training. The results plotted in Fig. 17 show misrecognition with a probability of 1% on the average from single aspect view when 60% or more views have been used for training.

Perfect Recognitions: The probability of misrecognition can be made negligible and even reduced to zero in two ways. One way which we describe here is to use more than one aspect view for a given target in interrogating the network and use a majority decision rule to decide the outcome. The multi-aspect views for recognizing aerospace targets in a practical target identification system could be readily collected and presented to the network as targets fly by the system. The training procedure for recognition from multi-aspect views remains the same as that used for recognition from single aspect view. Shown in Fig. 18 is the performance of the same network of Fig. 15 for recognizing the first group of targets

from three, rather than one, aspect views after the network has been trained with the available training set of aspect views. The three aspect views are randomly selected from the test set (100 views) and are sequentially fed into the network; the outputs from the network afterwards will give three labels whose majority vote determines the recognition outcome. There were 33 groups of three aspect views randomly formed from the total 100 aspect views and this ensures that almost every aspect view is included in the test. The correct recognition percentage in Fig. 18 is with respect to the 33 groups formed. It is seen from Fig. 18 that overall performance of the network improves by a factor of about 10% for recognition using three views over using a single view for interrogation and the correct recognition performance increases much faster as the percentage of the views used for training increases. The network now reaches 100% correct recognitions when 25% of the views for the Boeing 747 and 35% of the views for the B-52, respectively, are used for training. The network has also been tested with the second group of targets which was formed, as mentioned earlier, by adding a space shuttle scale model to the first group of targets. The network has been trained similarly using certain percentage of the total available aspect views from all three targets. The performance of the network in recognizing the second group of targets after it has been trained is shown in Fig. 19 where correct recognition performance of the network for the space shuttle is seen to be similar to that for the Boeing 747 airplane. From a practical standpoint it makes more sense to evaluate the performance of the net using multiple aspect views as test signals and majority vote when the three aspect views are successive or adjacent to each other rather than being distributed over a wide range of aspect angles. The situation is representative of probing the net with three successive frequency responses collected from the target as it changes its aspect relative to the measurement system because of relative motion. Such evaluation has also been done in our research and the performance of the network when the three aspect views are successive or adjacent to each other was found to be similar to the cases shown in Fig. 19 when the three aspect views are randomly selected and is therefore not shown. Recognition using multi-aspect views may be supported by biological vision system in which multiple perception fields are formed[19]. The second approach for reducing the misrecognition probability which we only mention here is to use multisensory information for both training and interrogation. For example polarization sensitive sensors can be used to measure the frequency response of the target for orthogonal polarization and data generated in this fashion can be used for both training and interrogating the network to enhance the

probability of correct classification.

Dynamic Range and Noise Considerations: There are two issues relevant to identification with neural networks which should be mentioned. The first issue concerns the dynamic range of input signals to the network. In applying neural networks to practical problems, it is usual to use binary digital inputs[7] or normalized inputs[21]. The range of inputs to the network shown in Fig. 15 is not constrained (is not binarized or normalized). It is the raw frequency response of the target measured for a given aspect corrected for range-phase and measurement system response[3]. The network can be trained with signals of arbitrary amplitude and can be tested with signals of arbitrary amplitude. No normalization is needed for preprocessing. For example, the network has been trained with a training set of aspect views with maximum amplitude of 0.5 (arbitrary units) for the B-52 airplane and interrogating the network with test set of aspect views of maximum amplitude of either 1 or 10^6 (arbitrary units) for the B-52 airplane was found to give the same result. This practically significant behavior, which we attribute to the highly nonlinear nature of the network (see equations (36) and (37)), indicates that there is little constraint on the dynamic range of the test signals applied to the trained net.

The second issue concerns the performance of the network with noisy data. Data in our study were collected in our experimental imaging facility and the SNR in the data was about 15-20(dB). Results in Figs. 17-19 reflect this value of SNR of the training data and test data. The network has also been tested with signals with smaller values of SNR and this has been done by adding to the test data artificial Gaussian noise in accordance to the distribution shown in (33). This situation was taken to be crude representation of when the test data are collected under nonideal situation when vibrations and wind buffeting of an aircraft produce noisy frequency response measurement. The training data were still the original frequency response data collected in our anechoic chamber measurement facility and there was no additional noise added to this training data. It is seen from Fig. 19 that the network is able to perform 100% correct recognition of the three test targets when the network was trained with 40% of the available aspect views and tested with the test set of experimental data without additional noise added to it. During the training process, the output was mapped from the input as shown in (37). When noise was added to the test set to test the network trained with 40% of the aspect views of experimental data, the performance of the network was as given in Table 2 by the row beginning with $\theta = 0$ for the Boeing 747 plane and the performance of the network for the other two target models

SNR	1	2	3	4	5	6	7	8	9	10
$\theta = 0$	74	78	85	88	91	93	95	97	100	100
$\theta = 0.1$	94	100	100	100	100	100	100	100	100	100

Table 2: Percent correct recognition of Boeing 747 for two different values of the threshold θ.

was found to be generally similar and is therefore not shown. It is seen from Table 2 that the performance of the network deteriorates as SNR decreases, but the network is still able to furnish 74% correct recognition even with SNR=1 (i.e. SNR=0(dB)). The performance of the network for this severe noise case can be improved by changing the zero threshold in (37) to a finite threshold during the training process and by keeping the zero threshold during the test or interrogation stage. For this case, the output neuron state in (37) during the training process was replaced by;

$$o(i) = U[|\tanh(u_i)| - \theta] = \begin{cases} 1 & \text{for } \tanh(u_i) > \theta \\ 0 & \text{for } \tanh(u_i) < -\theta \end{cases} \tag{38}$$

where θ represents the threshold. The output neuron state during the test process is still given by (37) or by $\theta = 0$ in (38). For $\theta = 0.1$ in (38), the performance of the network for the Boeing 747 scale model is shown in the last row in Table 2; the network has been trained with 40% of the available aspect views data with no additional noise added to them. The improvement in performance because of the finite threshold is readily noted and in the low SNR range an improvement by roughly 20 percentage points has been achieved. As the threshold θ increases, the performance of the network with respect to noisy data improves. But for very severe noise case, such as SNR=1, it is hard to achieve perfect recognitions, since for high noise level the role played by thresholding becomes less effective.

Effect of Spectral Window: All results presented above are for frequency response data collected over (6.5-17.5)[GHz] band in 101 points. A question of practical importance is whether fewer data points or a narrower spectral window can be used to ease the data acquisition process without sacrificing target identification ability by the trained net. Studies have therefore been done to assess the effects of spectral bandwidth and the number of data points over the band on the performance of the network in identifying the given target models. We have done that in several ways. One way is to keep the spectral bandwidth fixed at

(6.5-17.5)[GHz] and decrease the number of data points over the band; this is equivalent to changing the sampling interval of the frequency response data. In doing so, the number of neurons in the input layer which represent the number of frequency points in each measured frequency response is decreased and so is the the number of hidden neurons which is, as mentioned earlier, equal to the number of neurons in the input layer. Another way is to keep the sampling interval unchanged and to choose a portion of the (6.5-17.5)[GHz] band as the new spectral band, which again decreases the number of neurons in input layer representing the number of data points. In this case, the location of the selected spectral band has also been studied and was found to have little effect on the performance of the network. In all the above cases, the following behavior was observed: (a) when the number of data points and the number of neurons in the input layer representing the input data points to the net is decreased, by either changing the sampling interval or choosing a smaller spectral band, the number of learning cycles taken by the net to learn or internalize the given aerospace target models increases; this may be explained by the fact that for every target the amount of information in the data sets presented to the net during training has been reduced as the number of input data points is decreased and thus it takes relatively longer time for the net to learn the underlying structure in the data presented to it and form internal representations of the targets; (b) when the number of input data points to the net is too small, the net cannot learn or form the internal representations. The learning process does not converge. The minimum number of data points for which the learning process is observed to diverge is 17. This specific number is the closest integer to 101/6 and is the factor by which the sampling interval of the frequency data over the band (6.5-17.5)[GHz] has been increased; (c) when the number of input data points to the net is decreased, the performance of net when tested generally deteriorates; there is no clean pattern of deterioration and the average percentage of deterioration is 5%. For example, when the frequency band was reduced to (10.5-15.9)[GHz] over which there were 50 data points as the input to the net and 40% of the available 100 aspect views data (frequency responses) over such a band were used for training the net, the performance of the net, when it was tested with the data over the given frequency band, in recognizing Boeing 747 is 94%; this can be compared with results shown in Fig. 19 in which the net was able to achieve 100% correct identification of the Boeing 747 when it was trained with 40% of the available views of 101 data points over the (6.5-17.5)[GHz] band and tested with aspect views over this frequency band. The performance of the net with narrow spectral band data can be improved by increasing the

percentage of available aspect views used for training. When the input frequency data to the first layer of the net consisted of 50 points over the (10.5-15.9)[GHz] band, then if the percentage of the available aspect views used for training the net was increased to 50%, the performance of the net in identifying the Boeing 747 model was found to improve to 99%.

The divergence, mentioned in the preceding observation (b), occurs when the number of input data points to the net, and hence the number of input layer neurons, is too small and so is the number of hidden neurons, which equals also the number of neurons in the input layer. From theoretic considerations of the mapping power of multi-layer networks[8],[15], it is gathered that any mapping can be accomplished through a network of the type shown in Fig. 15 provided that an adequate number of hidden neurons is used (see also cautionary arguments in epilogue in [13]). Therefore studies have also been done to see whether the network can converge and learn to form the internal representations of the targets by increasing the number of hidden neurons in the net even if the number of input data points is too small. As mentioned earlier, when the number of input (frequency response data) points over the (6.5-17.5)[GHz] band to the net is reduced to 17, the learning process by the net could not converge; in this case, the number of the hidden neurons was also 17. However, by increasing the number of hidden neurons by 4 to 21, the net is able to converge and learn the internal representations for the given aerospace target models. It should be pointed out that, since the Fourier transform mapping between hidden layer and input layer in the net of Fig. 15 is carried out according to the discrete summation given in (35), the number of hidden neurons does not have to be equal to the number of input layer neurons (see also equation (8)). This result supports the theory in [8],[15]. By increasing the number of hidden neurons further, the number of learning cycles required by the net to converge during training process is observed to have reduced. Once there are enough hidden neurons and the net is able to converge to learn the internal representations for the given aerospace target models, our study did not cleanly show the improvement in performance, that is, in correctly identifying the given target models, of the net when tested as the number of hidden neurons was increased further[21].

8 Classification, Identification and Cognition

The term target identification and target recognition are frequently used interchangeably in the literature, and we have done the same here. Actually there is an important difference

between the two terms. The network we have described in the preceding section is not cognitive. It can correctly identify which of the three targets, it has been taught, is responsible for the sensory signal (e.g. the complex frequency response) presented at its input by producing a correct identification label at its output. The net is shown to be robust, in that noisy versions of its training set data are also correctly classified by triggering the correct identifying label. This robustness aslo provides for a generalization capability, in that when presented with a data set belonging to a learned object but was not specifically among the training set, the network will be able to classify it correctly. Generalization is important because it means the net does not have to be trained on all data sets needed to represent the object as dictated by angular sampling considerations (e.g. the scattering pattern of a target of extent L must be sampled approximately every $\bar{\lambda}/L$ [radians] when $\bar{\lambda}$ is the mean wavelength of observation). Without proper precautions, these robustness and generalization features mean also that every input presented to the network, of the preceding section, will produce a response by triggering a label even when it belongs to a novel object, i.e. one that was not learned by the network. The network is therefore not cognitive in that it has no mechanism for determining whether a presented signal belongs to a familiar (previously learned) object or to a novel object. Cognitive capability is essential for proper interpretation and use of a classifier network's response and for possible triggering of other useful mechanisms such as, for example, learning a novel input and adding it to the repertoire of the net.

There are several ways for imparting cognition to a classification network. One is to train the network on every object it could possibly encounter in its environment in the course of normal operation. This approach may not however be practical as it could require major increase in the size of the network specially when the number of possible targets to be learned becomes very large. A second way for imparting cognition is to add detectors at the system sensory level that analyze the received signals to see whether they belong to the class of targets of interest or not. Usually inference rules and decision trees are needed to make such distinction and more than one sensing modality is often indicated (e.g. measurement of altitude, speed, bearing, size (radar cross section), polarization, etc.). A third way for making a network cognitive is to incorporate cognitive capabilities in designing the net from the outset[22].

9 Discussion

Extrapolation and reconstructions by neural networks through learning was discussed in the first part of this paper. The approach is different from traditional ones and provides a novel way for almost perfect extrapolation and reconstructions from partial frequency response information. The approach is seen to lead by logical extension to the problem of target identification through the use of label representations at the output layer for target identification in place of exact object functions reconstructed in the extrapolation problem. The focus in using neural networks for extrapolation and recognition is on the structure of networks as well as the learning taking place in the networks, and not on any particular computation carried out by a particular neuron. The networks have been set up by studying ill-posed problems and the equivalent roles played by hidden neurons and regularization functions. The number of neurons in the hidden layer of the resulting networks need not be equal to that in the input layer as in most nets presented here and this number can be increased at will. The synaptic connections from input layer to both the hidden layer and the output layer need not be fixed as was done in this study but can also be set up ultimately through learning to handle any reconstruction problem in which the available data and the object functions do not necessarily have a Fourier transform relation or when the relation is not certain or known. In our work the measured frequency response data and the object function (the real part of the Fourier inverse of the frequency response, i.e. the real part of the complex range profile of the target) form a Fourier transform pair. For practical application of the target identification concept presented in this paper, one envisions that a library of frequency responses of scale models of targets of interest is generated by measurements under controlled conditions in an anechoic chamber radar scattering measurement facility for all target aspects relevant to practical encounter scenarios between a radar system and the target. The data generated in this fashion would be "taught" to a layered net by training as we have described. To use such "trained nets" to identify the actual radar targets (that correspond to the scale models used) from data generated by a broadband radar systems in the field, attention to scaling issues would be given by invoking the principle of electromagnetic similitude[20]. In this fashion one hopes to avoid the tedious and costly task of forming libraries in the field using actual radar systems and cooperative target "fly-bys".

The number of neurons in the input layer of our learning networks is determined by

the number of available frequency samples. The relation between the number of functions which can be learned by the network and the number of neuron in the hidden layer is still an open question; However, the theoretically established claim[8],[15] for the mapping power of multi-layer neuron networks, taken together with the findings of this work, provide strong evidence in support of the use of layered networks for target recognition. Nonlinear mappings in layered network enable the network to form the desired reconstruction mapping region[15] to give robust reconstructions from partial and noisy frequency information. The application of these concepts to the problem of noncooperative radar target identification discussed here provides "convincing" evidence of the capability of neuromorphic processing in providing results not attainable by traditional signal processing techniques.

ACKNOWLEDGEMENT: This work was supported in parts with grants from SDIO Innovative Science and Technology Office through the Office of Naval Research, JTFPMO through the Jet Propulsion Laboratory and the Army Research Office.

References

[1] A.N. Tikhonov and V.Y. Arsenin, *Solutions of ill-posed problems*, New York: Wiley, 1977.

[2] J. Hadamard, *Lectures on the Cauchy problem in linear partial differential equations*, Yale University Press, New Haven, 1923.

[3] N.H. Farhat and B. Bai, "Echo inversion and target shape estimation by neuromorphic processing," *Neural Networks*, vol. 2, pp. 117-125, 1989.

[4] T.P. Burg, "Maximum entropy spectral analysis," presented at the 37th meeting of the society of Exploration Geophysicists, October 1967.

[5] A. Papoulis, "A new algorithm in spectral analysis and band-limited extrapolation," *IEEE Trans. on Circuits and Systems*, vol. CAS-22, pp. 735-742, Sept. 1975.

[6] R.W. Gerchberg, "Super-resolution through error energy reduction," *Optica Acta*, vol. 21, pp. 709-720, 1974.

[7] J.J. Hopfield, "Neural networks and physical systems with emergent collective computational abilities," *Proc. of Natl. Acad. Sci. USA*, vol. 79, pp. 2554-2558, April 1982.

[8] D. Rummelhart, G. Hinton, R. Williams, "Learning internal representations by error propagation," *Parallel Distributed Processing*, vol. 1, MIT press, Cambridge, 1988.

[9] P. Werbos, and J. Titus, "Beyond regression: new tolls for prediction and analysis in the behavioral sciences," Harvard University dissertation, 1974.

[10] D.B. Parker, "Learning logic," *MIT Technical Report,* TR-47, Center for Computational Research in Economics and Management Science.

[11] Y. Le Cun, "A learning procedure for an asymmetric threshold network," *Proceedings of Cognitiva,* vol. 85, pp. 599-604, 1985.

[12] N.H. Farhat, *et al.*, "Projection theorems and their application in multi-dimensional signal processing," in *Advances in Optical Information Processing,* G.M. Morris (Ed.), SPIE, vol. 388, pp. 140-151, 1980.

[13] M. Minsky and S. Papert, *Perceptrons,* MIT press, Cambridge, 1969.

[14] B. Widrow, *et al.*, "Layered neural nets for pattern recognition," *IEEE Trans. on ASSP.* vol. 36, pp. 1109-1118, July 1988.

[15] R.P. Lippmann, "An introduction to computing with neural nets," *IEEE ASSP Magazine,* pp. 4-22, April 1987.

[16] N.H. Farhat, "Microwave diversity imaging and automated target identification based on models of neural networks," *Proc. IEEE* vol. 77, pp. 670-681, May 1989. See also: N.H. Farhat, S. Miyahara, and K.S. Lee, "Optical analogs of two-dimensional neural networks and their application in recognition of radar targets," in *Neural networks for computing,* J.S. Denker, Ed. American Institute of Physics, AIP Conf. Proc. 151, New York, 1988.

[17] C.W. Chuang and D.L. Moffatt, " Natural resonances of radar targets via Prony's method and target discrimination," *IEEE Trans. Aero. and Electro. Systems,* vol. AES-12, pp. 583-589, Sept. 1976.

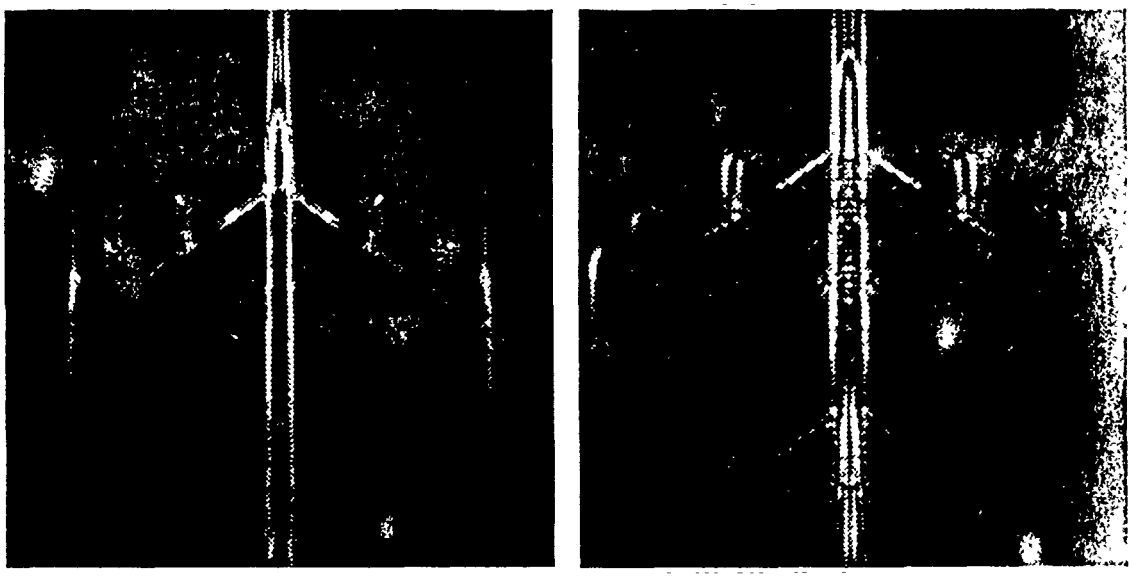
[18] M.L. Van Blaricum and R. Mittra, "Problems and solutions associated with Prony's method for processing transient data," *IEEE Trans. Anten. Propa.*, vol. AP-26, pp. 174-182, Jan. 1974.

[19] D. Hubel and T. Wiesel, "Receptive fields, binocular interaction and functional architecture in the cat's visual cortex," *J. Physiol.*, vol. 160, pp. 106-154, 1962.

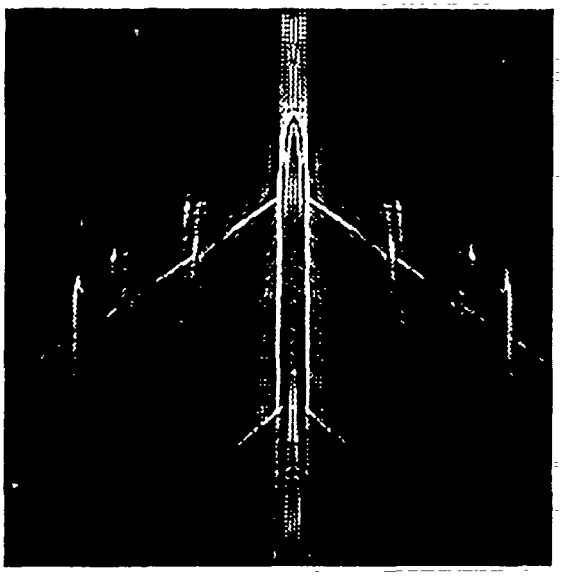
[20] J.D. Stratton, *Electromagnetic Theory*, McGraw Hill, New York, 1941, pp. 488-490.

[21] R. P. Gorman and T.J. Sejnowski, "Learned classification of sonar targets using a massively parallel network," *IEEE Trans. ASSP.* vol. 36, pp. 1135-1140, July 1988.

[22] S. Shinomoto, "A cognitive and associative memory," *Biol. Cybern.* vol. 57, pp. 197-206, 1987.

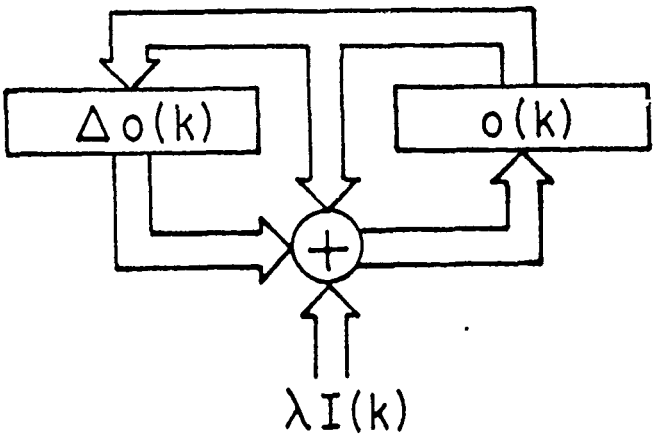


(a) (b)

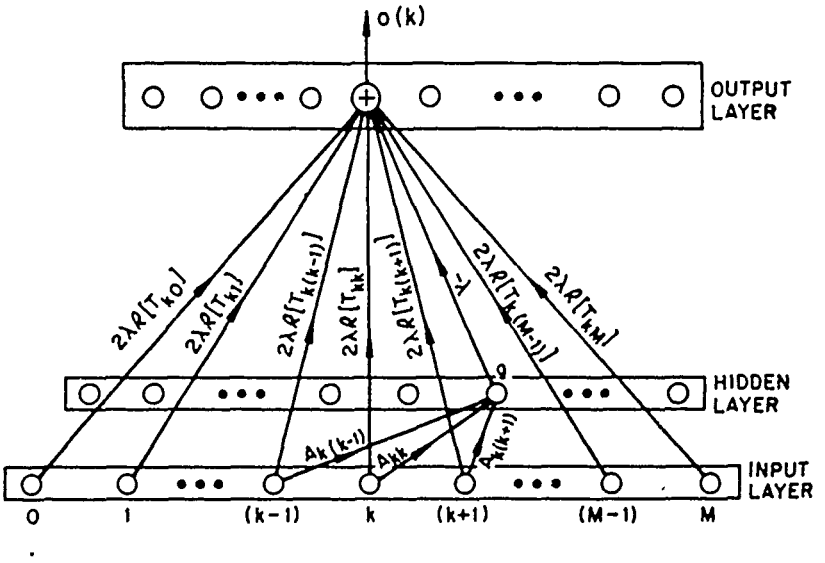


(c)

Figure 1: Microwave images reconstructed by DFT shown in (a) for spectral bandwidth (6-17)[GHz] and (b) for spectral bandwidth (2-26.5)[GHz]; (c) image reconstructed by non-linear neural net for the (6-17)[GHz] spectral bandwidth data.



(a)



(b)

Figure 2: (a) Realization of neural net processor; (b) realization of nonlinear regularization in neural net processor.

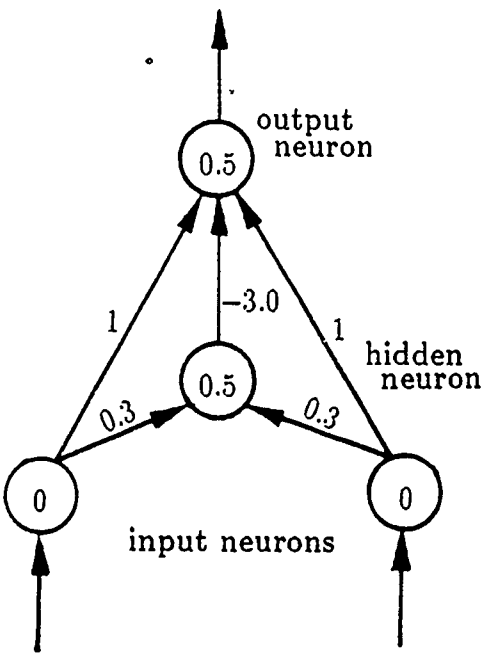
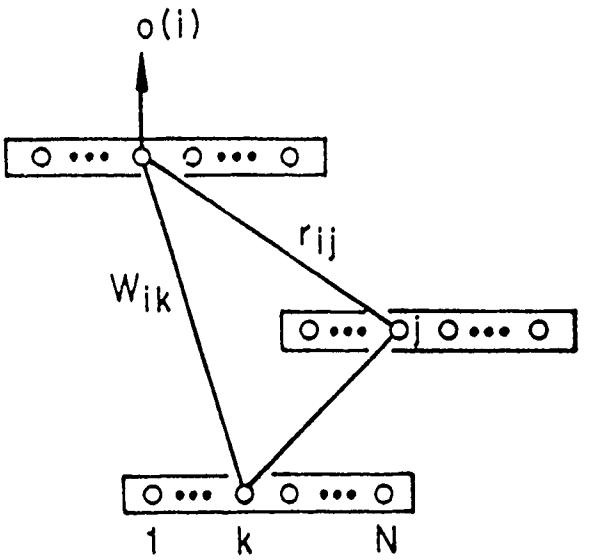
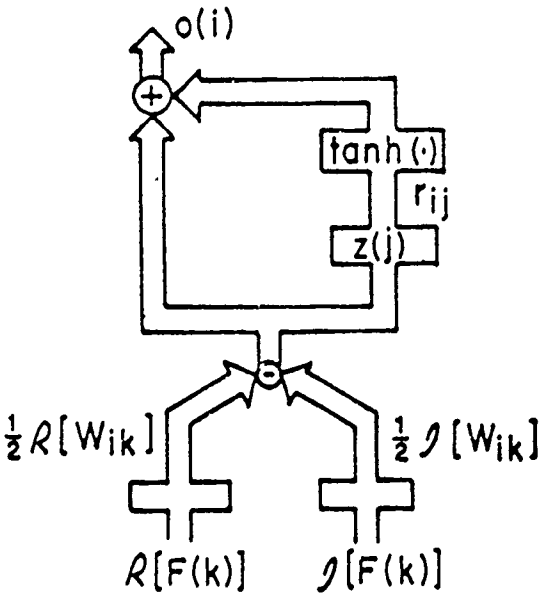


Figure 3: Network for XOR mapping.

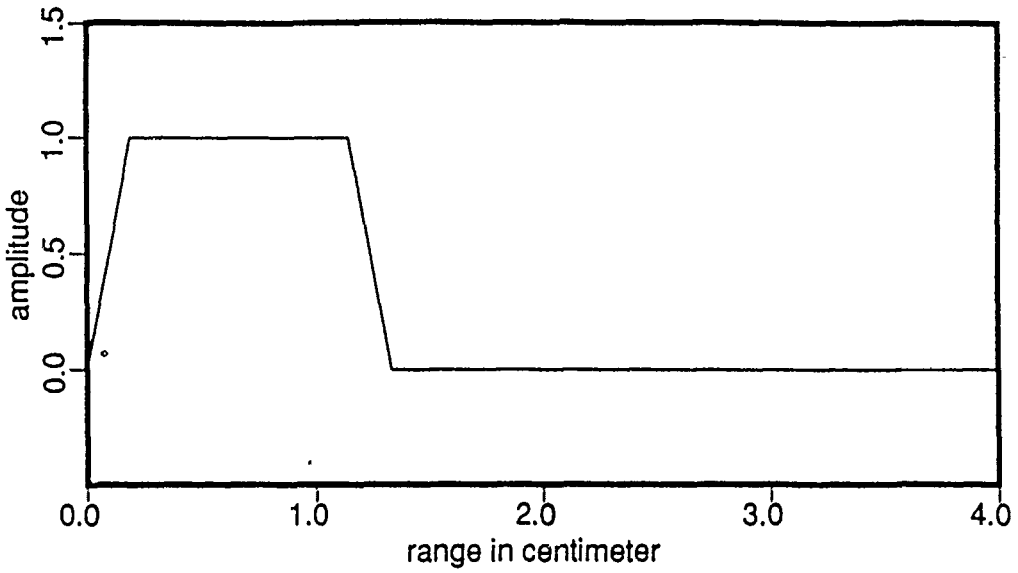


(a)

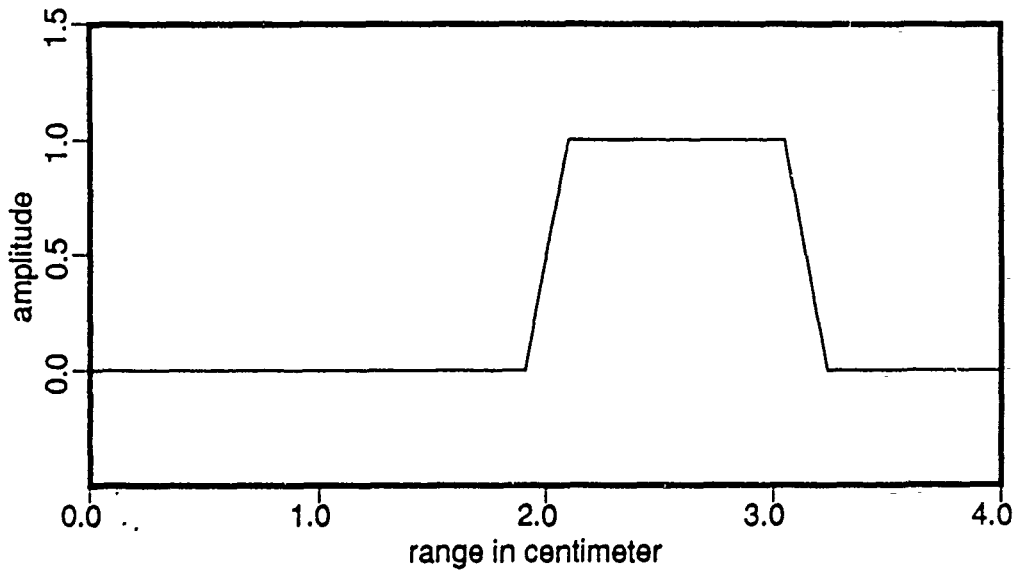


(b)

Figure 4: A three-layered neural net for reconstructions through learning. (a) Neuron distribution and connectivities; (b) equivalent flow chart.

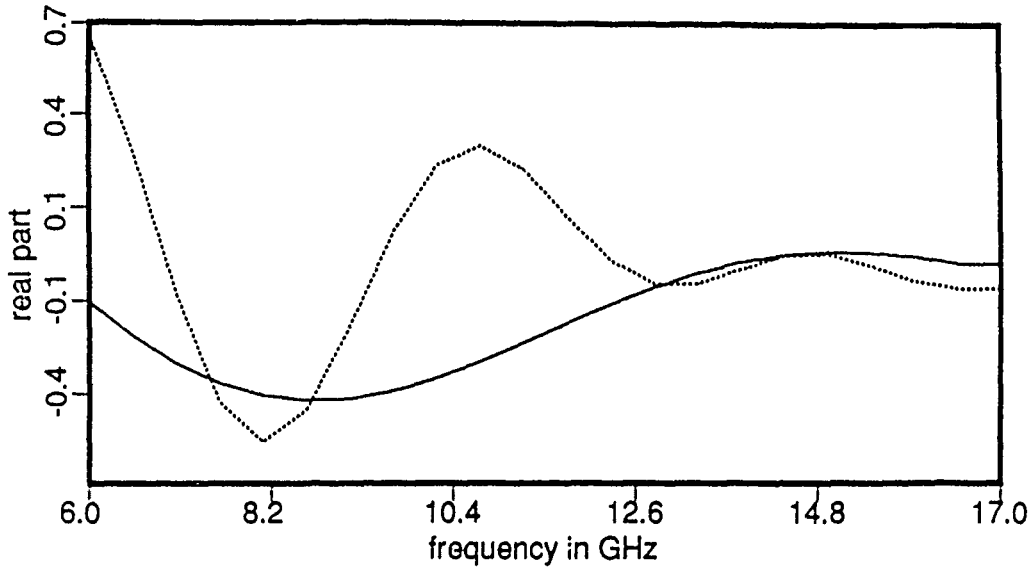


(a)

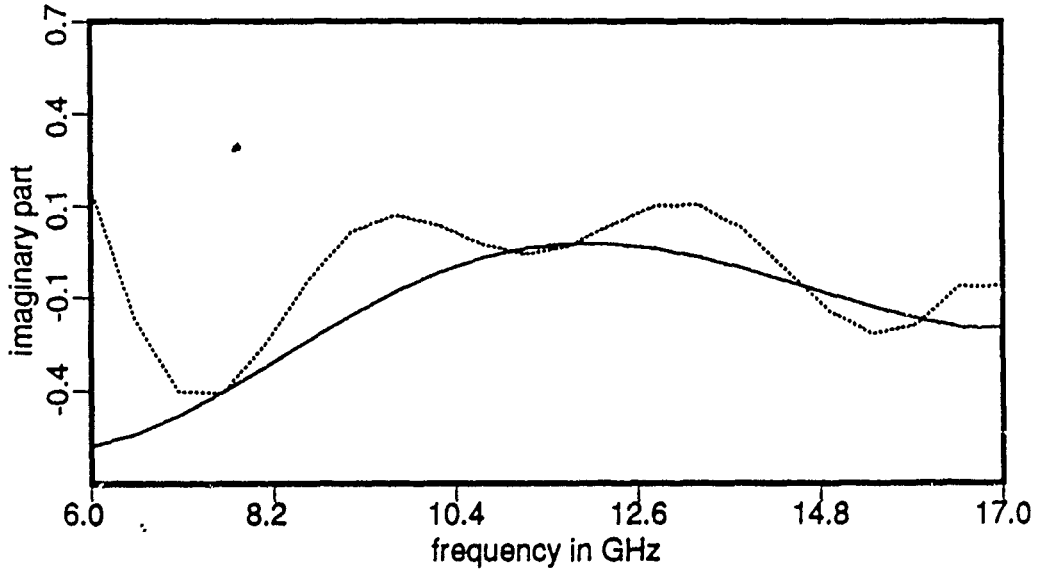


(b)

Figure 5: Two test object patterns $o(r)$ used in simulations: (a) first pattern; (b) second pattern.

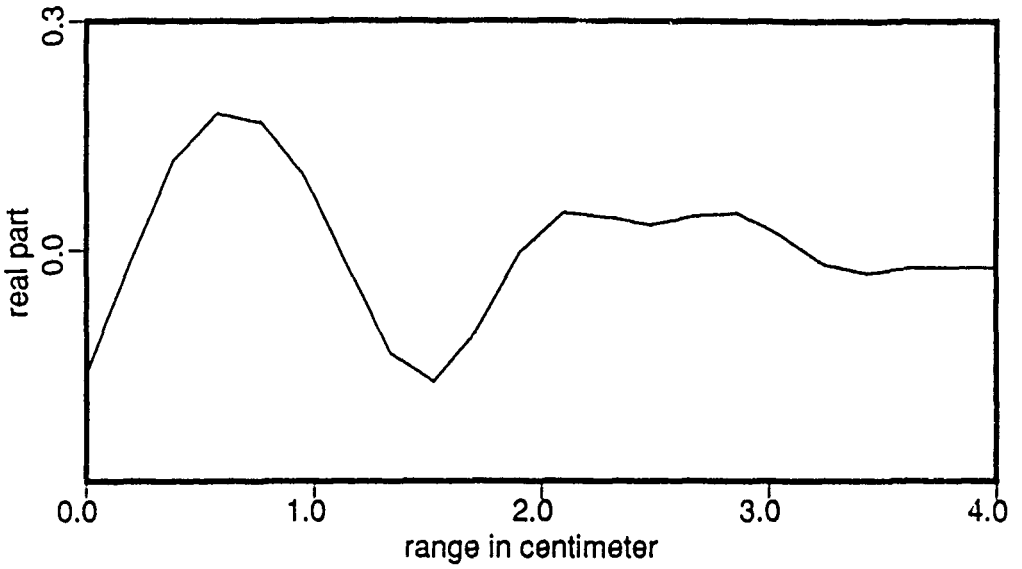


(a)

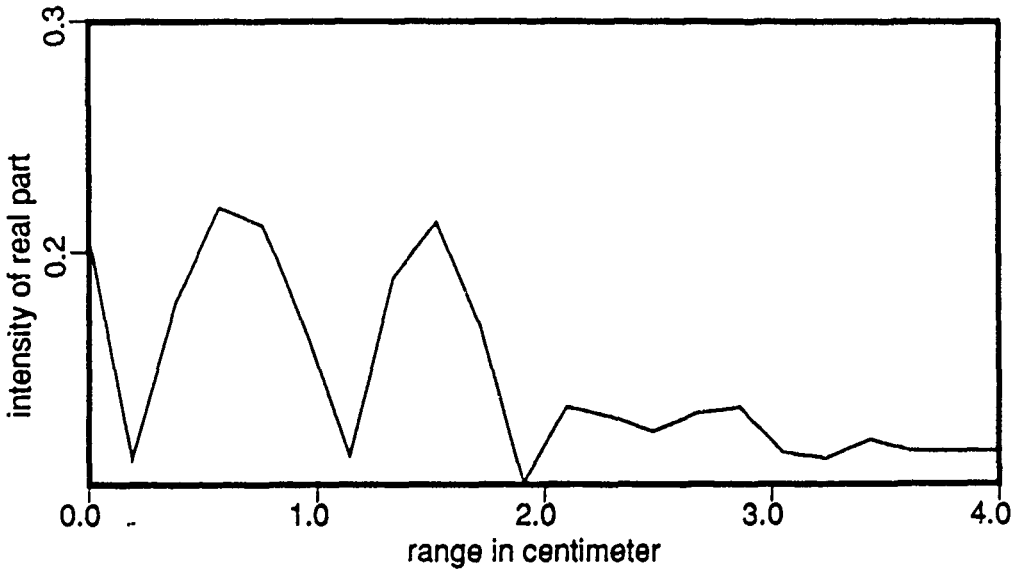


(b)

Figure 6: Frequency responses for the first object (solid line) and the second object (dotted lined): (a) real part; (b) imaginary part.



(a)



(b)

Figure 7: Reconstruction of the first object pattern by DFT: (a) real part; (b) intensity.

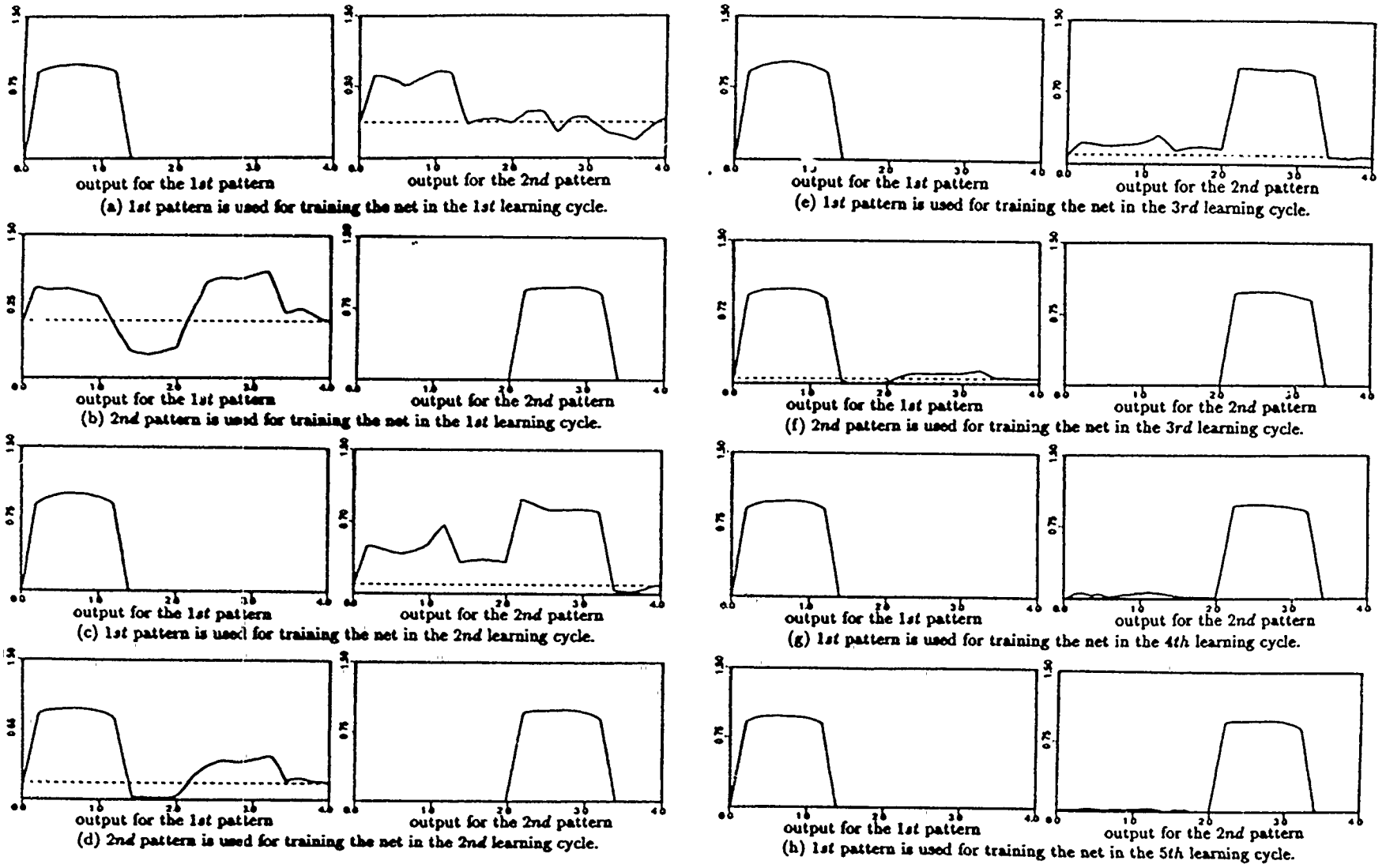
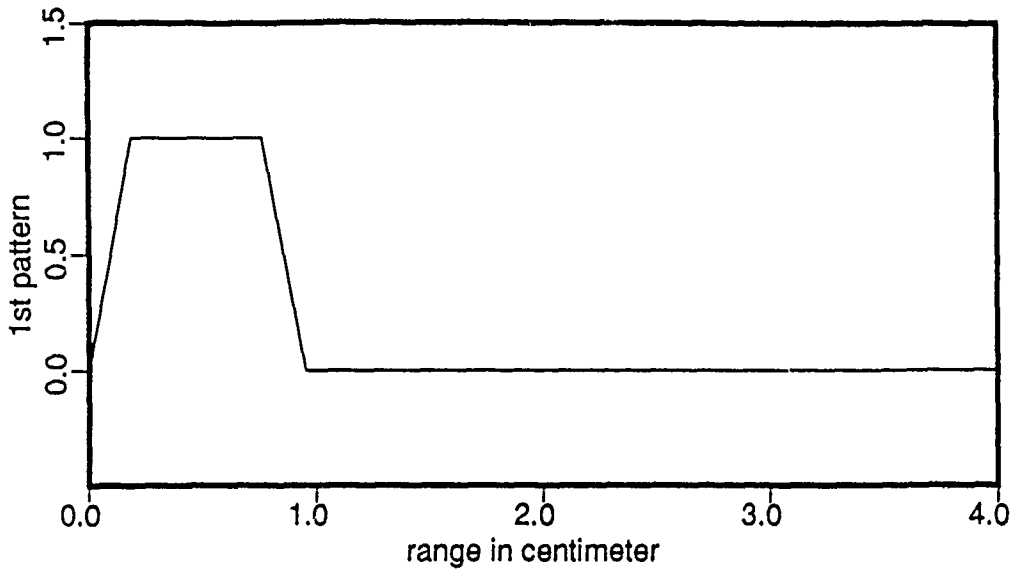
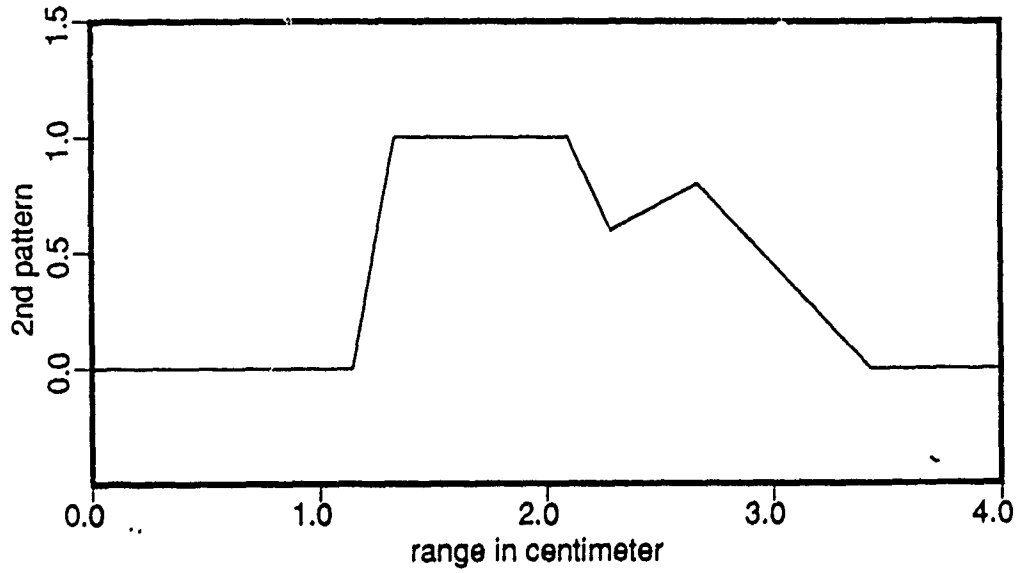


Figure 8: Sequence shows how the network gradually learns the synaptic connections to provide eventually the correct output for two patterns from partial associated frequency responses presented at its input.

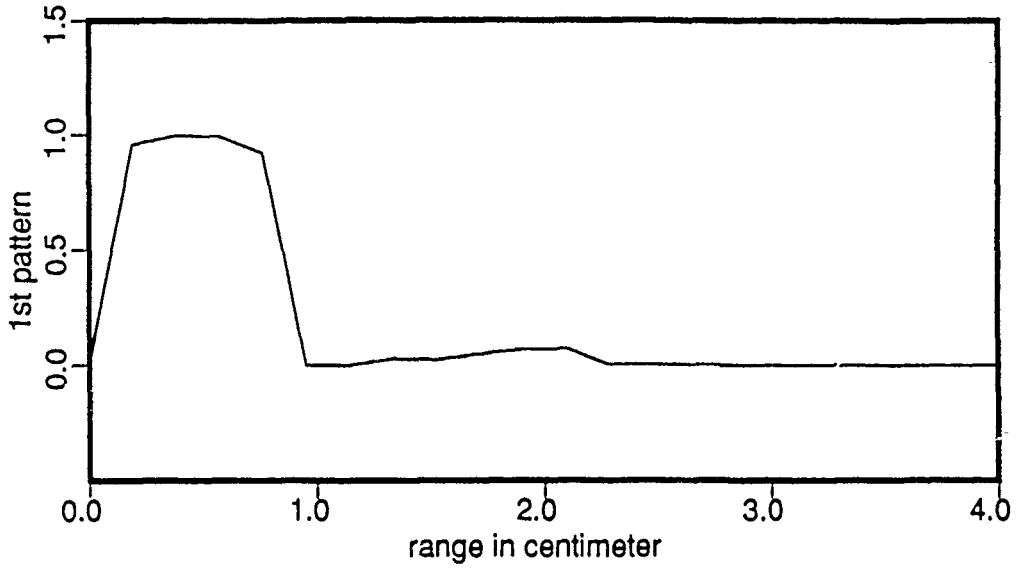


(a)

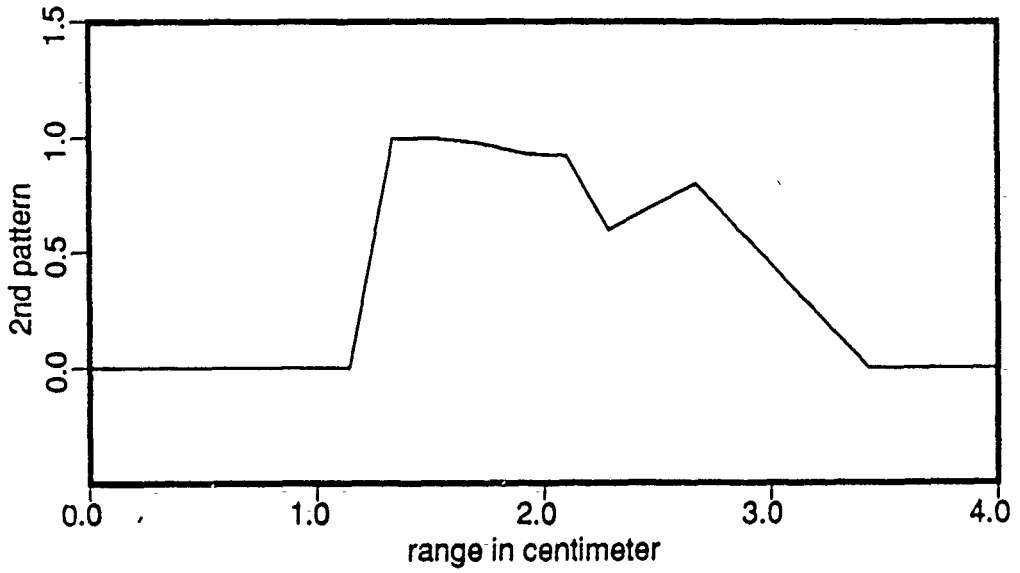


(b)

Figure 9: Object patterns with more complex shape used in simulation.

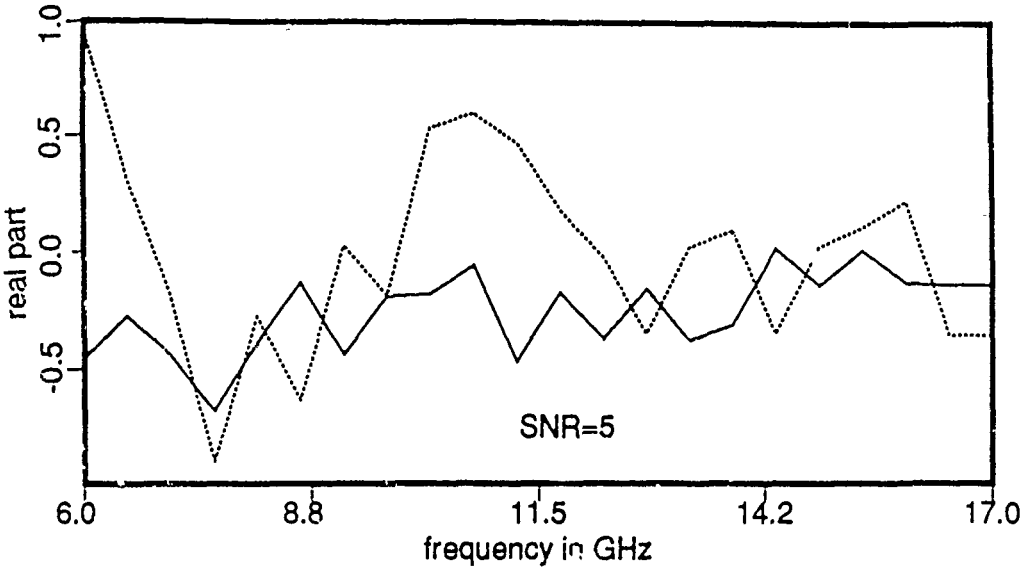


(a)

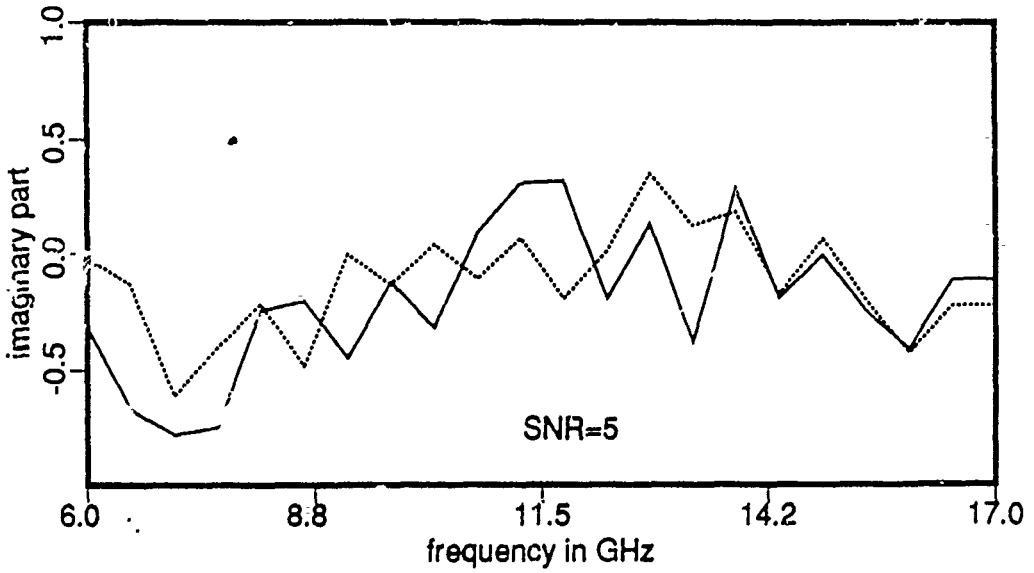


(b)

Figure 10: Reconstructions of the complex-shaped patterns of Fig. 9.



(a)



(b)

Figure 11: Noise contaminated frequency responses (SNR:=5) of the first pattern (solid line) and the second pattern (dotted line) of Fig. 5: (a) real part; (b) imaginary part.

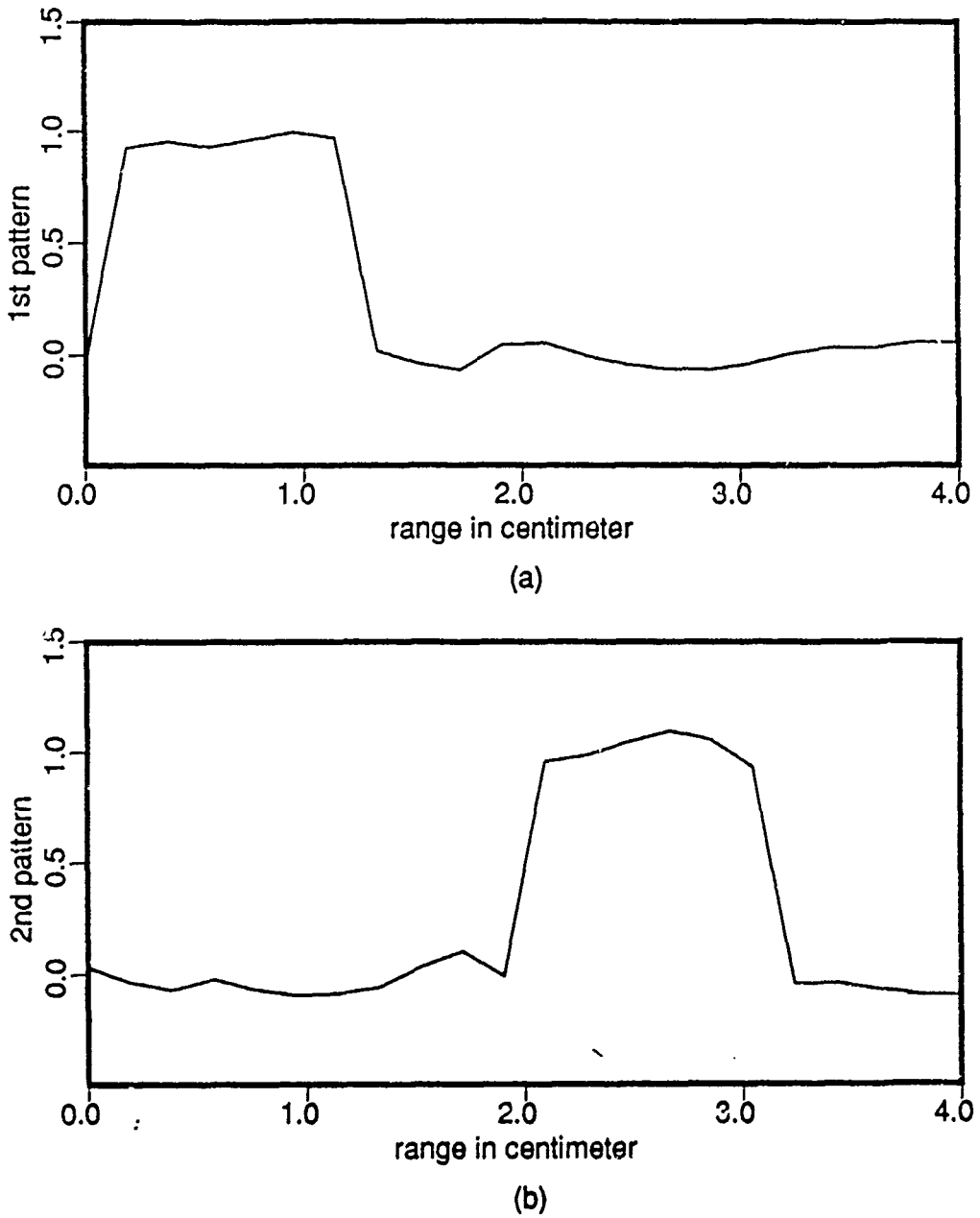


Figure 12: Reconstructions from noise contaminated frequency responses of Fig. 11: (a) first pattern: (b) second pattern.

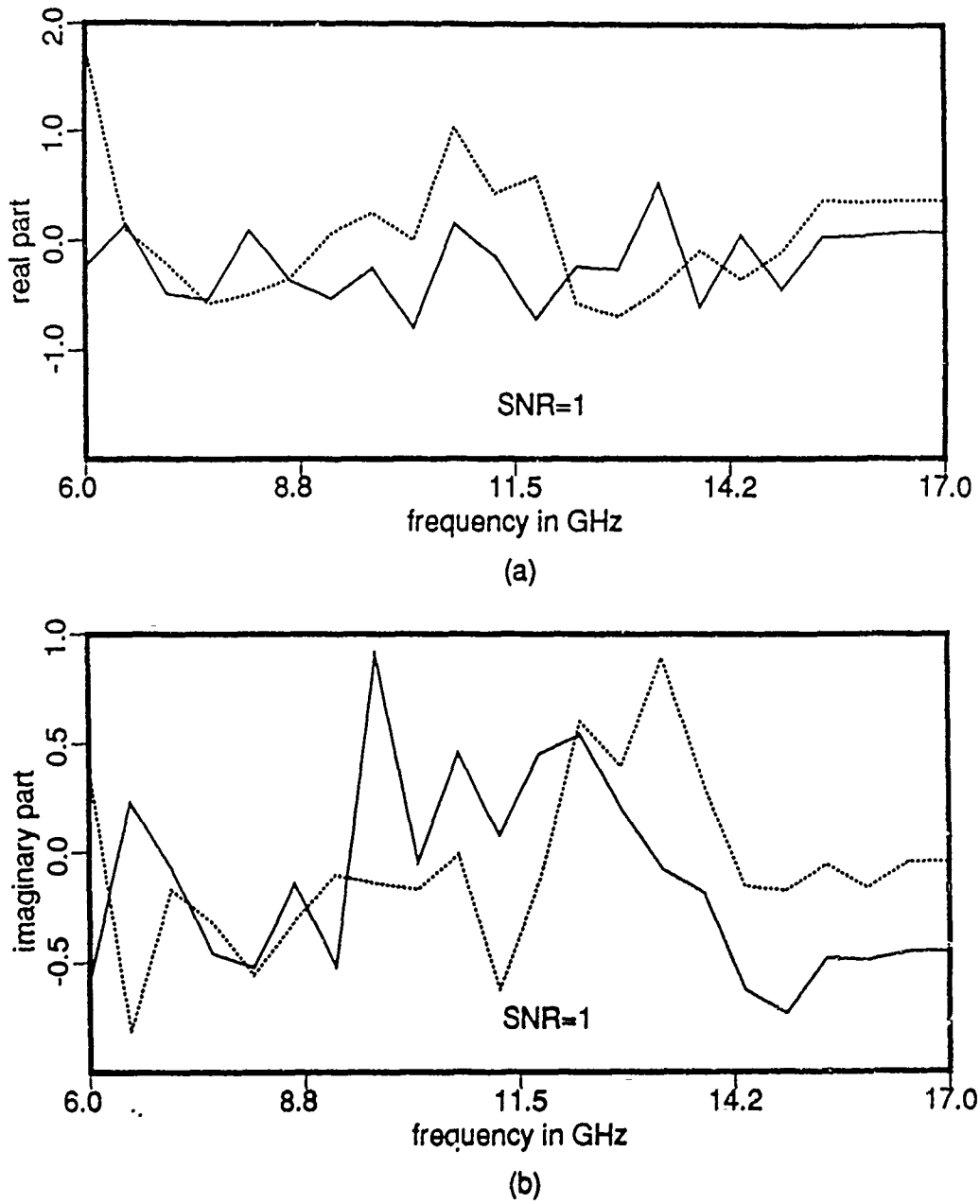
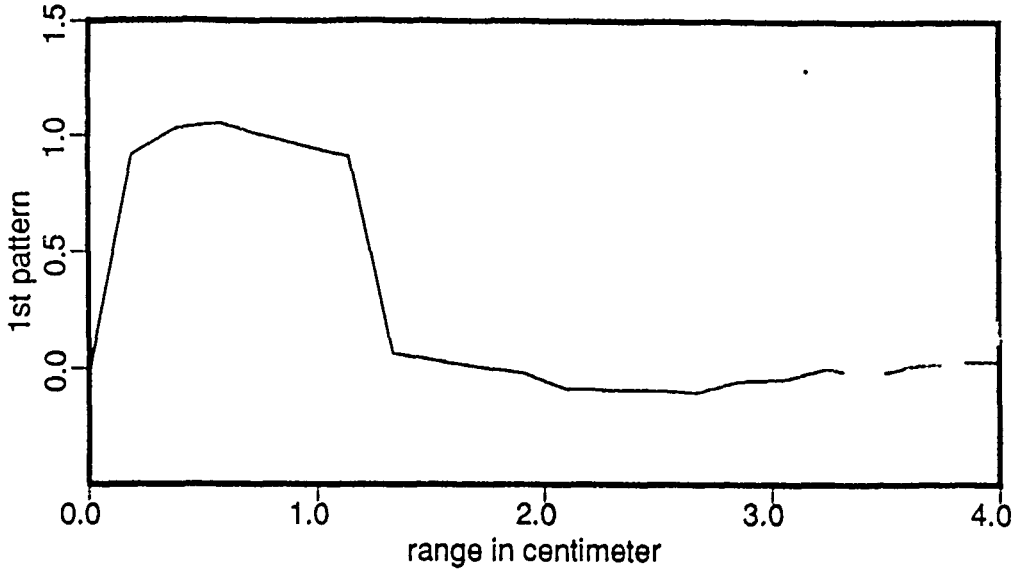
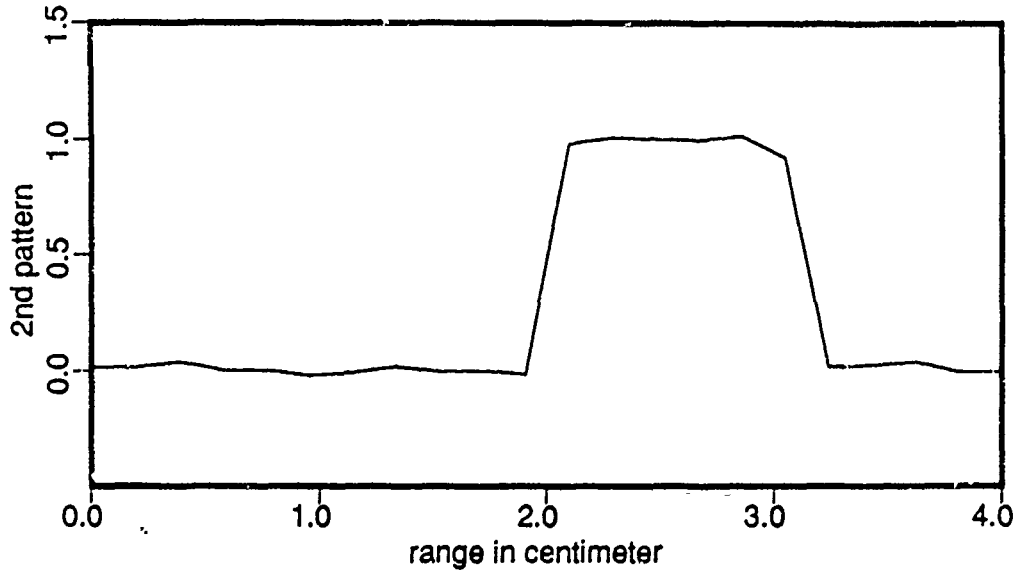


Figure 13: Noise contaminated frequency responses (SNR=1) of the first pattern (solid line) and the second pattern (dotted line) of Fig. 5: (a) real part; (b) imaginary part.



(a)



(b)

Figure 14: Reconstruction from the noisy data (SNR=5) of Fig. 11 after the network has been trained with instances of the noisy data (SNR=1) of Fig. 13 and the noise free data of Fig. 6: (a) the first pattern; (b) the second pattern.

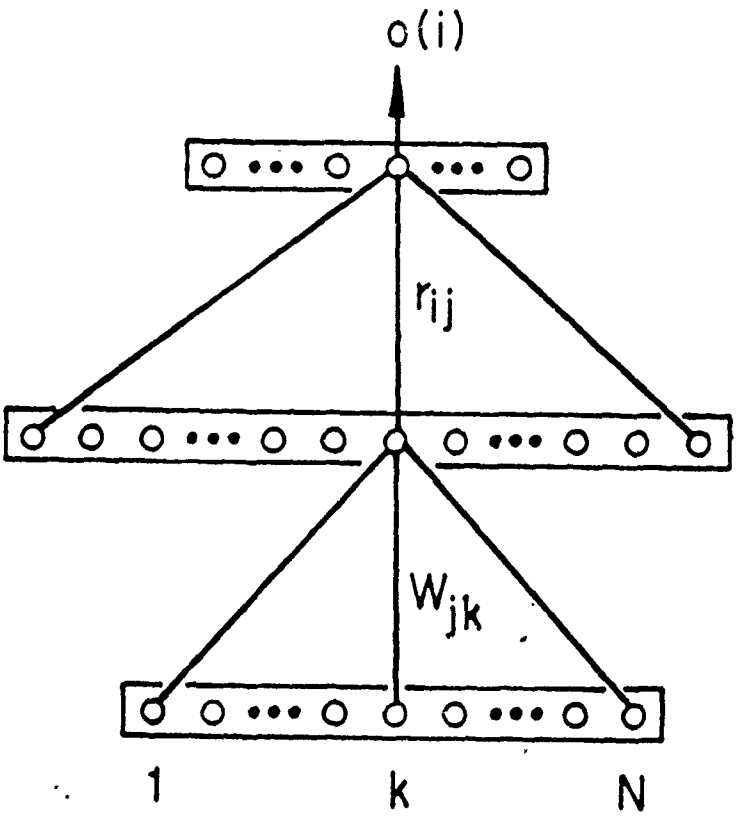
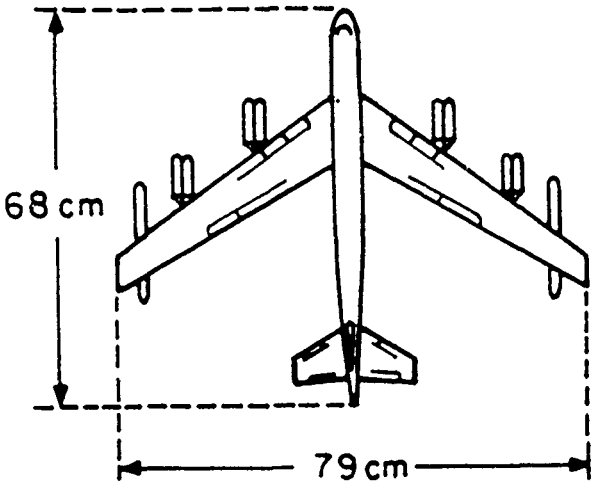
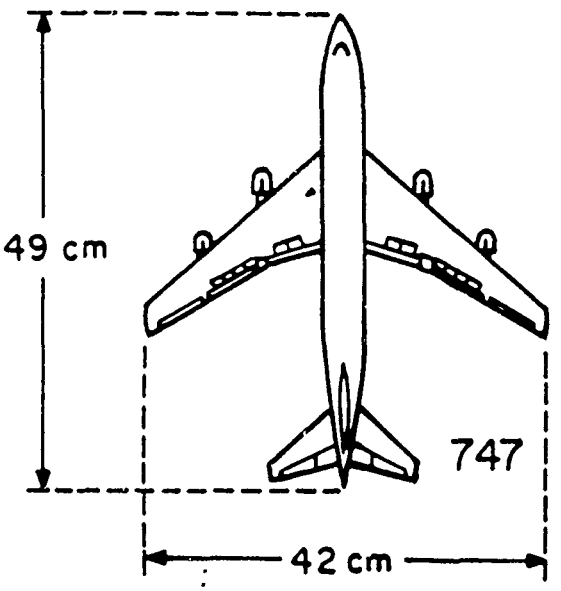


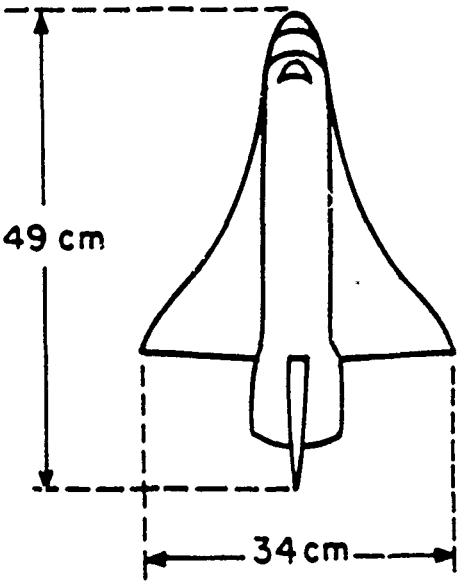
Figure 15: Neural network for target recognitions.



(a)



(b)



(c)

Figure 16: Three aerospace targets used: (a) a B-52 airplane; (b) a Boeing 747 airplane; (c) a space shuttle.

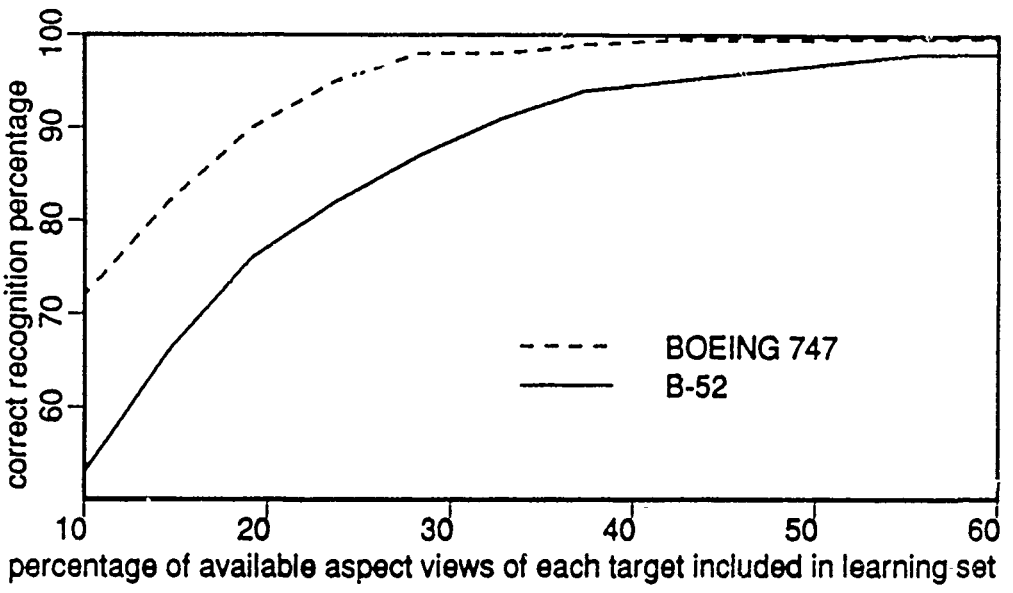


Figure 17: Correct recognition from single echo or "look" *vs.* size of training set for B-52 (solid line) and for Boeing 747 (dashed line).

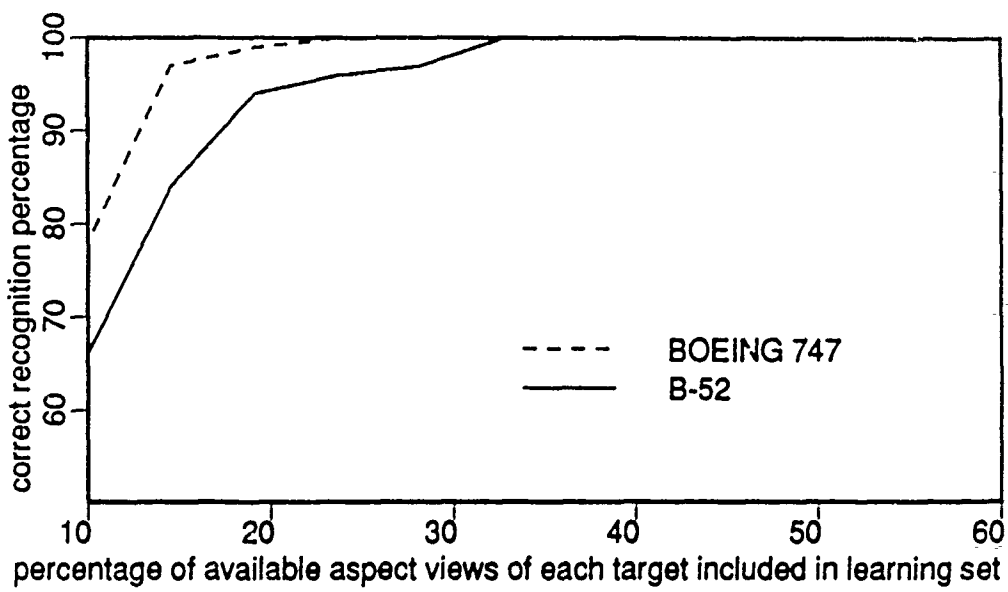


Figure 18: Correct recognitions from three looks *vs.* the size of training set when "a two out three" majority vote criterion is used, for B-52 (solid line) and for Boeing 747 (dashed line).

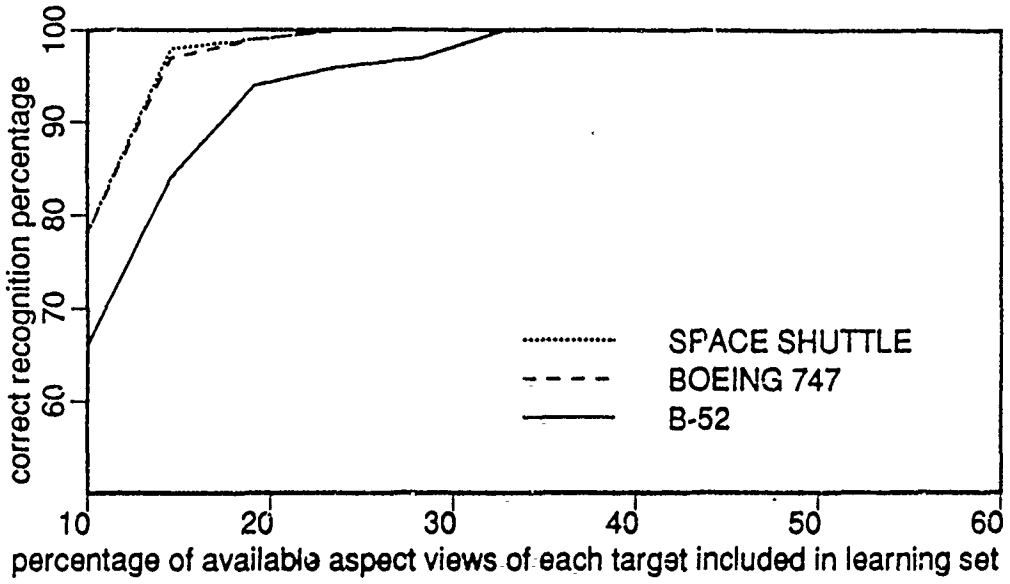


Figure 19: Correct recognitions *vs.* the size of training set when "a two out of three" criterion is used for correct classification for B-52 (solid line), Boeing 747 (dashed line), and Space shuttle (dotted line).

Appendix IV

1989 INTERNATIONAL SYMPOSIUM DIGEST
ANTENNAS AND PROPAGATION
VOLUME II
IEEE CATALOG NO. CH 2654-2/89
LIBRARY OF CONGRESS NO. 89-84327

PHASE SPACE ENGINEERING FOR NEURODYNAMIC TARGET IDENTIFICATION

N.H.Farhat* and H.Babri
University of Pennsylvania
The Moore School of Electrical Engineering
Electro-Optics and Microwave-Optics Laboratory
Philadelphia, PA 19104

Summary

Past research at the Electo-Optics and Microwave-Optics Laboratory [1]-[4] has led to inception and development of *microwave diversity imaging* where angular, spectral, and polarization degrees of freedom are combined to form images of complex shaped objects with near optical resolution. An example of attainable image quality is shown in Figure 1. This is a projection image of the scattering centers on a test object (a 100:1 scale model of a B-52). Co-polarized and cross-polarized data sets, each consisting of 128 azimuthal looks at the target extending from head-on to broad-side (90 degree angular aperture) and an elevation angle of 30 degrees with each look covering a (6–17) GHz spectral window were utilized in obtaining the image shown. Also a novel *target derived reference* technique [5] for correcting the frequency response data for undesirable range-phase (or range-phase time-rate (Doppler) when the target is moving) together with an *image symmetrization* [4] method were painstakingly developed and perfected before the image quality shown in Figure 1 could be obtained. In later discussion we will be referring to range-profiles of a target. The range-profile at a given target aspect is taken to be the real part of the Fourier transform of the corrected frequency response measured for that aspect. For a fixed spectral window and signal-to-noise ratio, the range-profile is independant of range and varies only with aspect.

Application of concepts and methodologies developed and demonstrated in the above research in practice would entail: either (a) the use of large, albeit sparse, recording imaging apertures to furnish the angular diversity needed or (b) the use of a single radar system that can track and interrogate a target, in the presence of relative motion, from different aspect angles in time to furnish the required angular diversity in an *inverse synthetic aperture radar (ISAR) or spot-light imaging mode.* The first approach is prohibitively costly specially when the target is remote and the angular aperture needed to achieve useful resolution is large. The second approach is non-real-time in nature as it requires observing the target over extended time intervals, and this may not be acceptable in numerous applications, in order to synthesize the required angular aperture. One is therefore constrained in practice to limited angular apertures or limited observation times and is therefore faced with the longstanding problem of image formation from limited and often sketchy (partial and noisy) information, i.e., one is faced with the classical problem of *super-*

resolution which has evaded a general solution for a long time. In other words, the problem is to recognize the target from a few looks

Among its many fascinating capabilities such as robustness and fault tolerance, the brain is also able to recognize objects from partial information. We can recognize a partially obscured or shadowed face of an acquaintance or a mutilated phatograph of someone we know with ease. The brain has a knack for supplementing missing information, based on previously formed and stored associations.

Here we propose and describe a new concept in automated radar target identification from a single "look" (coherent broad-band echo) based on neural net models. We view a neural net as a multidimensional nonlinear dynamical system capable of exhibiting powerful collective computational and signal processing functions that are fully and therefore best described by their phase-space behavior in terms of terminal, or periodic, or strange attractors and associated basins of attraction. The work

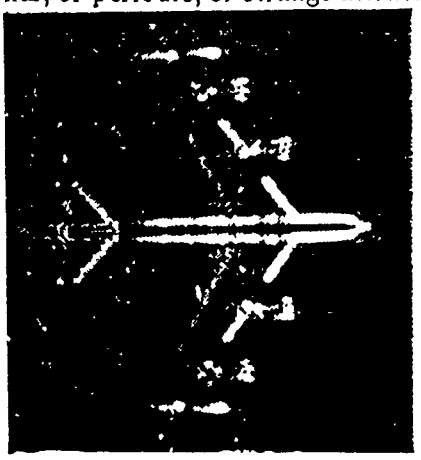


Fig.1: Microwave diversity image of a complex shaped object.

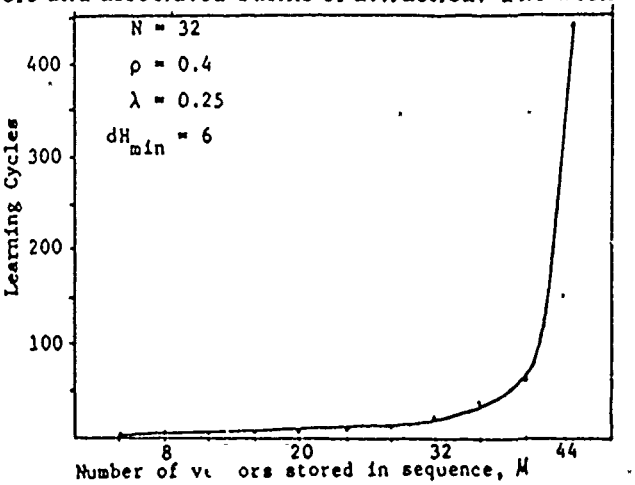


Fig. 2: Learning cycles needed for correct recall of a sequence of correlated vectors versus number of vectors, M in the sequence.

described is a direct extension of earlier work on neuromorphic target identification [6],[7]. We maintain that a central issue in understanding and applying neural nets today is finding ways for imparting to a net distinct phase-space behaviour that gives rise to desired functions. This *phase-space engineering* approach represents a totally new paradigm in signal processing that originates from known universal features of biological information processing in the nervous system. We will present initial results illustrating one possible method for applying the phase-space engineering concept to the longstanding problem of recognizing airborne targets from a single look. In this approach phase-space trajectories are formed from data contained in a library of range-profiles of the target within a prescribed "solid-angle of encounter" defined by all target aspects that can possibly be encountered by the radar system in typical practical situations. Initiating such a net from a state corresponding to any one of these profiles would trigger motion in its phase-space along the stored trajectory towards a terminal *"label" state* or *terminal attractor* that identifies the target uniquely.

To demonstrate this we choose a fully connected neural net of N bipolar binary neurons with state vector $s_i[1,-1], i = 1,2,\ldots,N$, (representing binarized representations of range-profiles), and connectivity matrix W with elements w_{ij}. The net updates its state vector synchronously according to $s_i = sgn(u_i)$, where $sgn()$ is the signum function and $u_i = \sum_j w_{ij}s_j$ is the action potential of the i-th neuron. Starting from an initial connectivity matrix $w_{ij}^{(0)} = 0$ the required connectivity matrix is formed by

$$w_{ij}^{(k)} = w_{ij}^{(k-1)} + \lambda(s_i^{(m)}s_j^{(m+1)} - s_i^{(m)}o_j^{(m)}) \qquad (1)$$

where $w_{ij}^{(k)}$ is the connectivity matrix at the end of the k-th learning cycle, $s_i^{(m)}$ is the m-th state vector in a sequence of M vectors, $o_i^{(m)}$ is the state vector arrived at after one iteration from an initial state $s_i^{(m)}$, λ is a positive real parameter controlling the learning process. In the above notation $s_i^{(M+1)}$ is taken to be a label vector identifiying the sequence. A learning cycle consists of repeated application of the above formula until the change $\Delta w_{ij} = w_{ij}^{(k)} - w_{ij}^{(k-1)}$ becomes sufficiently small such that subsequent testing of the formed net by initiating it from any member vector of the stored sequence results in its sequencing through all following members and terminating at the label vector. The number of learning cycles needed to learn M vectors using the above procedure is shown in Figure 2 for a neural net of $N = 32$ neurons (similar behavior was observed for $N = 64$ and 128). The stored sequence consisted of vectors with parameter ρ fixed at 0.4 (roughly similar behavior was observed for $0.1 < \rho < 0.9$). The Hamming distance between any pair of vectors in the sequence ranged from 6 to 9 ($6 < dH < 9$), thus the vectors in the sequence were well correlated. The value of the learning rate parameter λ was 0.25. It is seen that the training procedure or learning algorithm converges rapidly as long as $M \lesssim N$. As M increases beyond N the learning time, measured in number of learning cycles is seen to increase exponentially. It is worth noting however that a sequence with $M \gtrsim N$ can still be stored provided that a longer learning period can be tolerated. The sequential storage and recall capabilities exhibited here exceed by far the storage capabilities of Hopfield-like nets [8] where entities are stored as limit points in phase-space rather than in trajectories or orbits as is the case here. To evaluate the performance of the net, with w_{ij} obtained by the above training procedure, it was initiated from randomly selected phase-space points and its subsequent motion in phase-space from iteration to iteration observed. We find such random probing to be a useful tool in phase-space engineering work whereby qualitative information is obtained about the strength and nature of the basins of attraction of a terminal attractor and on whether it possesses spurious basins of attraction or not. Ninety such probing vectors whose Hamming distance from any of the vectors stored in the phase-space trajectory exceeded a given minimum distance dH_{min} were used. For $dH_{min} = 1$ or 2 not a single probing vector triggered the net in its trajectory. The scheme presented appears therefore to discriminate well against initial states that do not belong to the stored object information. Simulations were also carried out to verify that several distinct terminal attractors with unique basins of attraction

can be formed in the same network. We have been able to store 3 such attractors with filamentary basins of attraction formed from anywhere from a total of 20 to 40 vectors with ease. The ideas presented here demonstrate the viability of the neurodynamical principles of object recognition from a single look. They have important implication for distortion invariant radar target recognition and have potential for obviating the need for costly radar imaging systems of the type required for remote target identification from formed images. An extensive research effort aimed at reducing the concepts presented here to practice is currently underway in our laboratory. Aspects of this program will be briefly discussed.

This research is being carried out under grants from ARO and ONR, and with partial support from NSF and JPL.

References

[1] N.H. Farhat, "Principles of Broad-Band Coherent Imaging," *J. Opt. Soc. Am.*, vol.67, pp.1015-1020, Aug. 1977.

[2] N.H. Farhat and C.K. Chan, "Three-Dimensional Imaging by Wave-Vector Diversity," *Proc. 1978 Int. Symp. on Acoustical Imaging*, A.Metherell (ed.), Plenum Press, New York, (1980), pp.499-515.

[3] C.K. Chan and N. H. Farhat, "Frequency Swept Imaging of Three-Dimensional Perfectly Conducting Objects," *IEEE Trans. on Ant. and Prop.*, Special Issue on Inverse Scattering, vol.AP-29, March 1981, pp.312-319

[4] N.H. Farhat, *et al.*,"Prospects for three-dimensional projective and tomographic imaging radar network," *Radio Science*, vol.19, no.5, pp.1347-1355, 1985.

[5] N.H. Farhat, *et al.*, "A Target Derived Reference for Frequency Diversity Imaging," Poster paper presented at the *North American Radio Science / IEEE Meeting*, Quebec, Canada, 1980.

[6] Super-Resolution, Patent disclosure filed by University Patents Inc., Westport, Ct. on behalf of the University of Pennsylvania, April 1987.

[7] N.H. Farhat, "Microwave Diversity Imaging and Automated Target Identification," to appear in *Proc. IEEE*, Special Issue on Radar Cross-Section of Complex Shaped Objects, R.Stone (Ed.). (1988).

[8] J.J. Hopfield, "Neural Networks and Physical Systems with Emergent Collective Computational Abilities," *Proc. Natl. Acad. Sci., USA.*, vol.79, pp.2554-2560, (1982).